CLUBE

Z80

Novembro/84

N.º 26

Coordenadores: MARIA IRENE • ALEXANDRE SOUSA • J. MAGALHÃES • AV. BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

No interior:

Folheto Mercado Z80

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & C.ª, Lda./Porto

Tiragem: 500 exemplares, Novembro 1984

EDITORIAL

Nem sempre o nosso jornal, ou «Boletim», ou «Revista» publica um Editorial.

O EDITORIAL justifica-se quando existe qualquer coisa que implica uma mudança de rumo ou uma dificuldade na orientação que se deve imprimir a estas páginas ou ainda uma chamada geral, que por sua vez implica uma resposta de quem lê.

De quando em quando, toca o alarme, quando as listagens dos sócios que terminaram a assinatura do nosso jornal, atingem um índice superior a 40 %, sem que façam a respectiva renovação.

Não é demais chamar a atenção para o facto de que esta publicação não possui fins lucrativos. Quando muito, ficaríamos felizes se as despesas da sua edição fossem sempre cobertas pelas assinaturas dos sócios.

Tentem divulgar a existência deste jornal pois só assim fica assegurada a continuidade deste trabalho.
Coloca-se sempre a questão: Será que os assinantes são também leitores?

Tudo o que nós publicamos, tem real interesse?

Será que uma vez ou outra devemos modificar completamente, o formato ou orientação deste jornal mensal?

Na última expedição, verificamos que cerca de 160 assinantes, teriam de renovar a assinatura, e sempre que este desafio aparece, nós pessoalmente trememos pela continuidade do jornal. Continuamos a acreditar que é possível fazer qualquer coisa no campo das publicações apenas com a certeza de que os custos serão pagos pelos assinantes e não pela publicidade. Se não quer continuar a assinar o nosso jornal e portanto a fazer parte deste grupo de gente diferente, diga-nos a razão. Talvez procedendo assim, nos ajude a encontrar o caminho certo!

O melhor meio de contactarmos os dois .... quem orienta e aquele que lê .... é através de uma mensagem; não esqueça que nós precisamos de saber o que lhe interessa mais receber nestas páginas .... por favor escreva!

Alexandre Sousa

OS COLABORADORES DO JORNAL DO CLUBE Z 80

Tornar o Clube Z 80 mais activo, passa necessariamente, por colocar alguns dos que assinam esta publicação, numa posição diferente do simples espectador (leitor).
Tempos atrás, entregamos uma cassete, oferecida por gente amiga de VISEU, com alguns programas dedicados aos Rádio-Amadores, a um dos nossos sócios e amigos, devido aos insistentes pedidos desse nosso amigo (um abraço ao Victor Lima), de que gostaria de colaborar em algo de útil. Cinco meses depois, ainda continuamos esperando, que os célebres programas possam ser publicados...
O Clube Z 80, assina várias revistas internacionais, e possui imensa literatura que podia ser traduzida e adaptada, para publicação nas nossas páginas.
Estamos inteiramente abertos, a que vários dos nossos amigos e amigas, assumam a responsabilidade de traduzir, testar e depurar de lixo, uma montanha de material que pode ser publicado, e assim melhorar o conteúdo do nosso jornal. Deste modo, se TU, membro do Clube Z 80, TE QUERES RESPONSABILIZAR por uma dada secção periódica ou não, tens as nossas páginas abertas, e poderás receber a partir de agora, material (páginas fotocopiadas), em que existem temas e programas, que deverão ser publicados.
Deverás dizer-nos em que secção ou página específica, queres colaborar, escreve-nos a dizer se preferes os textos em Francês ou Inglês, e qual a periodicidade com que queres receber os textos de apoio.
Secções ou Títulos possíveis são:
LINGUAGEM MÁQUINA — JOGOS — AVENTURAS — ESTRATÉGIA — EDUCAÇÃO
HARDWARE — ROBÓTICA — RÁDIO-AMADORISMO — BASIC — PASCAL — FORTH — C
INFORMÁTICA MÉDICA — INFORMÁTICA APLICADA À PRODUÇÃO
etc., etc., etc.

Ficamos a aguardar as tuas notícias, as tuas ideias e a tua disponibilidade.

Alexandre Sousa

INTRODUÇÃO À LINGUAGEM MÁQUINA — ZX81/SPECTRUM

Autor: FERNANDO PRECES
SACAVÉM

(Cont. dos números anteriores)

PARTE III — COMO FUNCIONA O Z80

4.3. — As mnemónicas do z80 (Continuação)

Neste caso, a indicação do carry significa subtrair 1 ao resultado, quando usada a instrução apropriada (SBC - Subtração com carry). Para manipular esta instrução é preciso, porém um certo cuidado.

Veja com atenção os ensaios respectivos.

1.º Subgrupo — Instruções SUB

Este conjunto de instruções destina-se somente aos registros **simples** e tendo como interveniente principal o registro A. Assim a mnemónica SUB foi simplificada referenciando apenas o segundo interveniente.
Antes da operação indicada por uma destas instruções, o flag N é passado a 1 (indica ao Z80 que terá de haver uma subtracção) e os restantes flags, irão refletir o estado do Acumulador, após a operação.

Mnemónicas	Códigos	Tempos
SUB N	214, N	
SUB A	151	
SUB H	148	
SUB L	149	
SUB B	144	Iguais aos indicadores nas instruções ADD
SUB C	145	
SUB D	146	
SUB E	147	
SUB (HL)	150	
SUB (IX + d)	221, 150 + d	
SUB (IY + d)	253, 150 + d	

Ensaio 1:

Mnemónicas	Códigos	Comentários
LD HL, 27100	33, 220, 105	Apontador
LD A, 100	62, 100	
LD (HL), A	119	Coloca A, no endereço apontado
LD A, 255	62, 255	
LD E, 50	30, 50	
SUB E	147	(A — E)
SUB 100	214, 100	(A — N)
SUB (HL)	150	(A — (HL))
LD B, 0	6, 0	Resultado transferido para o BASIC
LD C, A	79	
RET	201	

(R: 5)

Exemplifica-se neste ensaio a introdução de 3 instruções deste subgrupo.

Ensaio 2: Uso incorrecto duma instrução SUB

Mnemónicas	Códigos	Comentários
LD A, 20	62, 20	
LD E, 30	30, 30	A subtração é incorrecta porque E > que acumulador
SUB E	147	
LD B, 0	6,0	
LD C, A	79	
RET	201	

Resultado esperado — 10
» obtido — 246

Analisando o estado dos flags antes e depois da operação SUB, verifica-se que o Z80 foi **avisado** do resultado não ser o correcto, e que tomou providências para poder reconhecer o resultado real. Não o fêz, porque a sequência de instruções que o poderia solicitar, não foi introduzida pelo programa.

Estado dos flags antes da operação SUB (posição 1) e após a operação (posição 2).

Flags	S	Z	X	H	X	P/O	N	C
posição 1	0	0	1	0	1	0	1	0
» 2	1	1	0	0	1	1	0	1

Que nos dizem então os flags?
Neste caso apenas nos interessa a análise dos flags (S) e (C).

1) Flag C — O carry-flag indica que foi ultrapassado o mínimo acumumulável no registo interveniente.
2) Flag S — O sinal-flag indica que o número contido no registo interveniente é negativo e que se encontra sobre a forma de **complemento por dois.**

Ora nós já falamos sobre o 2.º complemento aritmético e as suas formas de representação.
Sabemos que esses números estão compreendidos entre − 127 a + 128 e que o 8.º bit indicará o sinal respectivo.
Então:

bit de sinal

$$246_{(10)} = 1\ 111\ 0110_{(2)}$$

Se retirarmos 1 unidade ao número binário,

$$1110110 - 1 = 1110101$$

e negarmos o resultado;

$$\overline{1110101} = 0001010$$

verifica-se que o resultado real é (−) 10; ou ainda, por um processo muito mais rápido:

$$246 - (1 * 2 \uparrow 8)$$
$$= 246 - (1 * 256) = -10$$

A forma de forçar a máquina a devolver ao Basic um resultado **negativo,** envolve instruções de que ainda não falamos, bem como a manipulação das rotinas de cálculo em numeração com ponto flutuante, situadas na ROM, que estudaremos no próximo capítulo.
Utilizando o retorno ao Basic após a operação, e nalguns casos as instruções SBC, poderá o leitor para já contornar o problema.
Vamos ver como.

2.º Subgrupo — As instruções SBC

Também aqui o flag N é comutado para 1 e os restantes flags reflectirão o estado do registo interveniente após a operação.
Os tempos de execução são iguais aos das instruções ADC (ver grupo 6).

Mnemónicas	Códigos
SBC A, + N	222, + N
SBC A, A	159
SBC A, H	156
SBC A, L	157
SBC A, B	152
SBC A, C	153
SBC A, D	154
SBC A, E	155
SBC A, (HL)	158
SBC A, (IX + d)	221, 158, + d
SBC A, (IY + d)	253, 158, + d
SBC HL, HL	237, 98
SBC HL, BC	237, 66
SBC HL, DE	237, 82
SBC HL, SP	237, 114

Já aqui foi afirmado, mas repete-se dada a sua importância, que **apenas** o registo A ou o par HL foram concebidos no Z80, para efectuarem operações aritméticas ou lógicas.
Na elaboração dum programa em C/M deve-se assegurar a cada momento uma certa disponibilidade destes registos, pois é necessário ter em conta que um computador numérico executa todo o tipo de manipulação binária à base de cálculo.

Ensaio 1: Subtracção em que o maior número se encontra no registo BC e o menor em HL.

RAMTOP em 27.999

```
LD  BC, 57000    1, 168, 122  } Colocação dos registos
LD  HL, 27000   33, 120, 105  } nas condições propostas

AND       A        167         Carry reset "0"

LD A, C            121      LOW BYTE de BC PARA A
SUB L              149      subtrai com LOW BYTE de HL
LD C, A             79
LD A, B            120      HIGH BYTE de BC para A
SBC A, H           156      subtrai os HIGH BYTE com CARRY
LD B, A             71
RET                201

          PRINT   USR    28000
           (Resultado: 30000)
```

Ensaio 2 — Subtrair B do conteúdo de HL.

```
LD HL, 29000    33, 72, 113  } Colocação dos registos
LD (HL), 200        54, 200  } nas condições
LD B, 100            6, 100  } propostas
LD A, (HL)          126
SUB B               114
LD (HL), A          119
RET                 201

       RANDOMIZE    USR    28000
            PRINT   PEEK 29000

           (Resultado: 100)
```

Ensaio 3 — Subtracção cujo resultado é um valor negativo.

Conforme já mencionei este processo é **meramente demonstrativo,** pelo facto de exigir um retorno ao Basic ao fim de cada subtracção.

```
AND A                   167
LD A, 20             62, 20
LD E, 30             30, 30
SUB E                   147  } número negativo E > A
                             } em complemento por 2
LD (29000), A   50, 72, 113
SBC A, A                159    (A - A) - carry = 255
LD (29001), A   50, 73, 113
RET
RANDOMIZE USR 28000
LET N = PEEK 29000
IF PEEK 29001 = 255 THEN GOTO (x)
PRINT N
STOP
(x) LET K = 1 * 2 ↑ 8 [(1 * 2 ** 8) para o ZX81]
    PRIN N - K                    (RESULTADO: - 10)
```

(Continua no próximo número)

HARDWARE

Autor: Alexandre Sousa

Pretende-se nesta coluna, ir transmitindo um tipo de experiência diferente.
Diferente do jogo simples ..... tecla 5 esq ..... tecla 6 ..... descer ..... tecla 7 ..... subir ..... tecla 8 ..... direita.
Nem todos os que abrem as páginas do Clube Z 80, querem o jogo, nem todos os que o procuram querem ou estão preocupados com o Hardware. Não defendemos que no meio está a virtude, mas sabemos que hoje queremos escrever sobre o Hardware, portanto quem tiver unhas que se agarre ..... !
INPUT/OUTPUT: o «hardware» é o mundo dos homens do ferro de soldar, dos que conhecem os fios, os circuitos integrados, e dos que são chamados quando queremos ligar o computador ao mundo externo.
Ligar ao mundo externo significa ligar o computador ao televisor, à impressora, ao termómetro, ligar ou desligar o alarme, receber a indicação do electrocardiógrafo ou comandar o semáforo da rua.
Tudo isso é INPUT/OUTPUT ou seja .... entrada/ /saída. ....
Qualquer computador, independentemente do fabricante, pode processar milhares de operações por segundo. Se não existe comunicação com o exterior, o seu uso é verdadeiramente limitado. Portanto, quando existe Input/Output existe a ligação autentica com a resolução de problemas. Input/Output é o termo usado para descrever a ligação com o mundo externo ao computador. Para discutirmos ou falarmos de entradas e saídas, teremos de conhecer o computador (ele mesmo!), e partir desse ponto para conhecermos o modo como ele comunica com o exterior.
De um modo geral, podemos dizer que o computador, possui dois blocos fundamentais: PROCESSADOR e MEMÓRIA.
O processador (verdadeiro motor do sistema!) segue uma determinada sequência de instruções que vai fazer com que sejam manipulados dados, ou seja irá apresentar esses dados de uma determinada forma, tratados ou alterados em relação à sua forma original.
A sequência de instruções e os dados serão guardados na memória para uso (e abuso) do processador.
Três tipos de ligações eléctricas, chamadas BUSES, são usadas pelo PROCESSADOR para estabelecer a comunicação com a MEMÓRIA:
ADRESS BUS (bus de endereços) .... liga o processador a uma posição determinada.

DATA BUS (bus de dados) .... comunica dados de e para o processador.
CONTROL BUS (bus de controlo) .... permite designar que e quem está em comunicação com o processador.

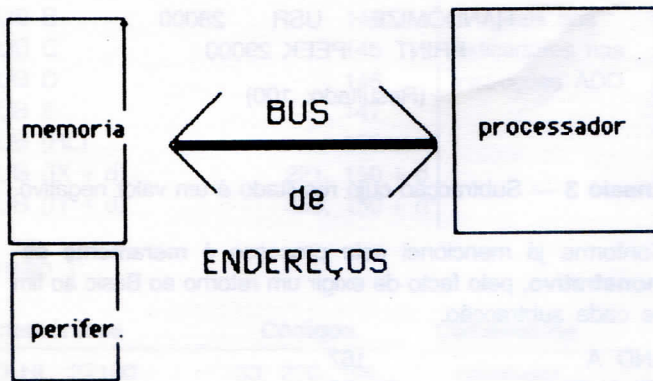



FIGURA 1

A memória de um computador está organizada em milhares de posições físicas, cada um das quais possui um endereço único e a possibilidade de guardar um dado ou uma instrução (são coisas diferentes!).
O processador pode ter acesso a toda a informação que está em memória, colocando o seu «pedido» no ADRESS BUS. Esse sinal (pedido) estará representado pelo endereço da posição de memória que interessa ao processador. Ele mesmo (processador) terá de especificar se pretende extrair informação (READ — ler —) ou se pretende colocar informação (WRITE — escrever —) nessa posição de memória.
Este tipo de sinal READ/WRITE é colocado no CONTROL BUS, o qual contém uma linha de sinalização que permitirá o sincronismo (actuar ao mesmo tempo), entre o processador e a memória.
Desde que exista esse sinal, poderá ser trocada informação entre o PROCESSADOR e a MEMÓRIA através do DATA BUS.
O Processador distingue entre DADOS e INSTRUÇÕES devido a um ciclo de temporização interno, ou seja, inicialmente (INSTRUCTION FETCH) procura instruções e executa as operações designadas nesse tipo de instrução.
Existe uma posição no interior do processador para reter a (ou as) instruções a executar no momento; esse registo chama-se contador do programa — PROGRAM COUNTER.
A instrução indicada pelo contador do programa pode ser a de ir procurar numa posição de memória, um dado e actuar sobre ele (modificando-o) ou, pelo contrário, escrever um dado numa posição de memória.

OPERAÇÕES DO COMPUTADOR

Para que o computador possa fazer correr um programa (RUN), o computador deve:

— ter acesso à memória para obter instruções
— ter acesso à memória para obter dados
— processar dados
— colocar os dados processados na memória

Podemos colocar duas questões:

a) Como obter programa e dados da memória?
b) Como obter os resultados do programa?

Resposta: através dos dispositivos de INPUT/OUTPUT (entrada/saída).
Um sistema completo, como estamos habituados a ver quando olhamos para um APPLE // ou para um IBM PC, não está composto apenas por um Processador e por uma Memória. Possui dispositivos separados: unidade central (processador e memória); Teclado; Monitor; Impressora e Drive para diskettes, tudo isto alimentado a 220 volts C.A.
O Spectrum, que geralmente gira à nossa volta, possui uma unidade integrada, — teclado, processador, memória — o monitor e o T.V. — e possui ou não impressora e gravador de cassettes (eventualmente: microdrive). Tudo aquilo que não é PROCESSADOR e MEMÓRIA INTERNA será considerado Mundo Exterior.
ATENÇÃO: a fita da cassette ou microcassette (waffer) e a película da diskette, são considerados como memória externa ou de massa!

COMO ESTÃO LIGADOS OS DISPOSITIVOS EXTERNOS AO CONJUNTO PROCESSADOR/MEMÓRIA?

Dois métodos podem ser usados:

O primeiro deles, e o mais simples, consiste em colocar os dispositivos no ADRESS BUS como se fossem posições de memória (a primeira vez que projectamos e construímos um microcomputador, usamos este método).

O processador pode enviar dados para ..... ou obter dados de ..... a partir dos periféricos (dispositivos externos) usando instruções do tipo MEMORY-REFERENCE INSTRUCTIONS, ou seja, instruções do tipo — referenciada como memória.

Este método denomina-se MEMORY-MAPPED I/0 porque designa os periféricos como uma parte da memória do Processador. Por exemplo os processadores 6800 e 68 000 da Motorola funcionam dessa forma.

VANTAGENS: Existem instruções do processador que funcionam simultaneamente para os periféricos e para a memória.

DESVANTAGENS: Não existe disponibilidade para usar a totalidade da memória endereçável pelo processador, quer para programa, quer para dados .... por outras palavras: O método de Memory-Mapped I/0 reduz a dimensão máxima de memória disponível ..... um processador de 8 bits, como os que estamos habituados a usar e que pode endereçar 64 000 possíveis posições de memória, possui uma perda de memória disponível, que não é de desprezar.

O processador INTEL 8080 e o ZILOG Z 80 usam um diferente esquema de trabalho:

Os dispositivos I/0 estão igualmente ligados ao processador através do MEMORY BUS, mas ..... existe uma instrução especial e um sinal no CONTROL BUS para tornar diferente o processo de entrada/saída.

O endereço de um dispositivo externo é diferente do endereço da memória, e assim torna disponível a totalidade da memória endereçável pelo processador.

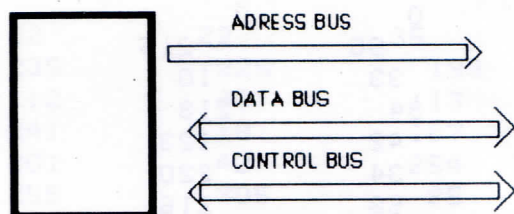



FIGURA 2

CONVERSOR ZX81 → SPECTRUM

O programa CONVERSOR DE PROGRAMAS ZX81 → SPECTRUM, de Fernando d'Almeida Preces, encontra-se à disposição daqueles que o pretenderem adquirir, gravado em cassete.

NOTA: Esse programa saiu listado nos n.os 20, 21, 22 e 23 do CLUBE Z80.

Preço: 500$00

MIRROR - Programa para duplicar microcassetes

SPECTRUM

Adapt.: JORGE AGUIAR
PORTO

Este programa, muito útil para duplicar os ficheiros guardados em microcassette, possui 3 partes:

a) Programa em BASIC que deve ser gravado tal como está na listagem publicada. A sua execução é iniciada automaticamente.

b) Screen$ — Trata-se de um écran (figura) que vai ser guardado na zona de memória conhecida como «Buffer da Printer».

c) CAT — Um bloco de código máquina que faz o catálogo dos programas invisíveis — os que são gravados com um nome que começa com o Chr$ 0.

Para o gravar não se esqueça de fazer o CLEAR 64 854.

```
  10 CLEAR 64854: LOAD *"m";1;"M
irrorSCREEN$ "CODE
  20 LOAD *"m";1;" CAT "CODE
  30 CLEAR #: LET z$="": RANDOMI
ZE USR 23400
  40 PRINT AT 19,0;"1-Leitura de
 cabecalhos";AT 20,0;"2-Catalogo
 de cartridges";AT 21,0;"3-Copia
s de programas"
  45 LET i$=INKEY$
  50 IF i$<"1" OR i$>"3" THEN GO
 TO 40
  55 GO TO VAL i$*1000
1000 GO SUB 9999
1010 PRINT AT 19,0;"Qual e o pro
grama desejado?"
1015 INPUT i$
1020 IF i$="" THEN GO SUB 9999:
GO TO 40
1030 IF LEN i$>10 THEN GO TO 101
0
1031 GO SUB 9999
1032 PRINT AT 19,0;" Coloque no
Microdrive 1 a""" microcassete e
m que esta o""" programa desejad
o.": PRINT #0; FLASH 1;"   -Prim
a uma tecla-  ": PAUSE 0
1033 GO SUB 9999
1040 GO SUB 8000
1044 LET a$="": FOR n=1 TO 9: LE
T a$=a$+INKEY$#4: NEXT n
1046 GO SUB 9999: LET n=CODE a$(
1)
1047 IF n=2 THEN PRINT AT 19,0;"
Ficheiro de dados:";i$: GO TO 10
20
1048 GO TO 1050+(n*10)
1050 PRINT AT 19,0;"Programa Bas
ic:";i$
1051 LET n=CODE a$(8)+256*CODE a
$(9): IF n<>65535 THEN PRINT "Au
to-execucao na linha ";n: GO TO
1120
1052 PRINT '"Nao se auto-executa
.": GO TO 1120
1060 PRINT AT 19,0;"Array numeri
co:";i$
1061 PRINT "Variavel: ";CHR$ (CO
DE a$(6)-32);"()"
1062 GO TO 1120
1070 PRINT AT 19,0;"Array alfanu
merico:";i$
1071 PRINT "Variavel alfanumeric
a:";CHR$ (CODE a$(6)-96);"$()"
1072 GO TO 1120
1080 PRINT AT 19,0;"Bytes:";i$
1081 PRINT "Inicio:";CODE a$(4)+
256*CODE a$(5)
1082 PRINT "Comprimento:";CODE a
$(2)+256*CODE a$(3)
1120 LET x=0: PRINT #0;"Prima um
a tecla.": PAUSE 0: GO SUB 9999:
 CLEAR #: CLOSE #4: CLOSE #5: LE
T z$="": GO TO 40
2000 GO SUB 9999: PRINT AT 19,0;
"Coloque no Microdrive 1 a";AT 2
0,0;"cartridge desejada."
2010 PRINT #0; FLASH 1;"  -Prima
 uma tecla-  ": PAUSE 0: LET z$=
""
2011 GO SUB 9999: PRINT AT 20,0;
"Espere um momento,por favor."
2015 POKE 23766,1: RANDOMIZE USR
 65506:
2016 GO SUB 9999
2020 PRINT AT 20,0;"Nome da cart
ridge:";z$( TO 10)
2030 GO SUB 9998
2040 LET z$=z$(13 TO )
2050 GO SUB 9999
2060 PRINT AT 20,0;"Nome:   ";z$(
 TO 10)
2065 GO SUB 9998
2070 LET z$=z$(12 TO ): IF LEN z
$<6 THEN GO TO 2100
2080 GO SUB 9999
2090 GO TO 2060
2100 GO SUB 9999: PRINT AT 20,0;
"Kilobytes livres:";z$(2 TO )
2110 GO SUB 9998
2120 GO SUB 9999: RANDOMIZE USR
23400: GO TO 40
2999 STOP
3000 GO SUB 9999: PRINT AT 20,0;
"Qual e o nome do programa?"
3010 INPUT i$
3015 IF i$="" THEN GO SUB 9999:
GO TO 40
3016 IF LEN i$>10 THEN GO TO 301
0
3017 GO SUB 9999: PRINT AT 19,0;
" Coloque no Microdrive 1 a""" m
icrocassete em que est£ o""" pro
grama desejado.": PRINT #0; FLAS
h 1;"   -Prima uma tecla-   ": P
AUSE 0
```

```
3020 GO SUB 9999
3025 CLEAR #: LET z$="": RANDOMI
ZE USR 23400
3030 GO SUB 8000: IF a=2 THEN GO
 SUB 3040: GO TO 3090
3035 GO SUB 3040: GO TO 3070
3040 MOVE #4 TO #14
3050 GO SUB 9999: PRINT AT 19,0;
"Coloque no Microdrive 1""a mic
rocassete em que deseja""gravar
 o programa.": PRINT #0;"Prima u
ma tecla."
3055 PAUSE 0: GO SUB 9999
3060 OPEN #5;"m";1;i$: RETURN
3070 DEF FN p(p)=PEEK p+256*PEEK
 (p+1)
3080 POKE FN p(5*2+23566+8)+FN p
(23631)+66,4
3090 PRINT #5;z$
3100 CLOSE #4: CLOSE #5
3110 PRINT AT 20,0;"Copia comple
ta.": PRINT #0;"Outra gravacao?(
s/n)": PAUSE 0
3120 IF PEEK 23560=115 THEN GO S
UB 3050: GO TO 3070
3130 LET z$="": GO SUB 9999: GO
TO 40
8000 PRINT AT 20,0;"Espere um mo
mento,por favor.": POKE 23766,1:
 POKE 23768,4
8010 FOR n=1 TO 10
8020 IF n>LEN i$ THEN POKE USR "
a"+n-1,32: GO TO 8040
8030 POKE USR "a"+n-1,CODE i$(n)
8040 NEXT n
8042 LET a=USR 23296: IF a=1 THE
N PRINT #0; FLASH 1;"Nao existen
te!": BEEP .7,50: PAUSE 50: GO S
UB 9999: GO TO 40
8050 RETURN
9998 PRINT #0;"Prima uma tecla p
/ continuar.": PAUSE 0: INPUT ""
: RETURN
9999 INPUT "": PRINT AT 19,0;TAB
 31;" ";AT 20,0;TAB 31;" ";AT 21
,0;TAB 31;" ": RETURN
```

Apos introduzir o programa Basic grave-o em Microdrive com a instrucao:

```
SAVE *"m";1;"run" LINE 1
```

Verifique a gravacao com a instrucao:

```
VERIFY *"m";1;"run"
```

Se a gravacao estiver boa, desligue o computador, torne a liga-lo e introduza o seguinte programa:

```
10 FOR n=23296 TO 23513
20 INPUT a
30 PRINT n,a
40 POKE n,a
50 NEXT n
```

Corra o programa e introduza a seguinte lista de numeros, da direita para a esquerda e de cima para baixo:

```
58   216  92
205  39   23
33   17   0
175  237  66
1    0    0
216  50   215
92   33   10
0    34   218
92   42   123
92   34   220
92   58   216
92   135  33
22   92   95
22   0    25
217  229  217
229  207  34
221  203  24
70   40   13
175  207  33
207  44   225
217  225  217
1    1    0
201  221  203
4    150  175
229  207  33
209  225  115
35   114  221
126  67   230
4    198  2
79   6    0
217  225  217
201  0    0
0    0    0
0    0    0
0    0    42
83   92   43
197  229  1
11   0    205
85   22   209
33   59   0
193  9    213
235  115  35
114  35   235
1    247  255
9    1    9
0    237  176
225  35   237
75   79   92
167  237  66
34   50   92
1    0    0
201  196  21
90   40   0
40   0    11
0    245  42
75   92   126
254  90   40
11   254  128
202  112  6
205  184  25
235  24   240
35   78   35
70   3    197
229  9    205
```

```
82          22          35
235          225          193
112          43          113
241          18          167
201          42          254
255          205          85
24          62          13
215          68          77
201          201          0
```

```
Grave o codigo maquina introdu-
zido com a instrucao:
SAVE *"m";1;"MirrorSCREEN$ "CODE
 23296,255

Verifique a gravacao com:
VERIFY *"m";1;"MirrorSCREEN$ "CO
DE

Se estiver bem gravado,desligue
o computador e introduza este
outro programa:

10 CLEAR 64854
20 FOR n=64855 TO 65535
30 INPUT a
40 PRINT n,a
50 POKE n,a
60 NEXT n
Introduza depois a seguinte
lista de numeros,da esquerda
para a direita e de cima para
baixo.
```

```
58          216          92
215          1          22
205          232          15
221          126          25
205          247          23
1          255          0
237          67          201
92          205          196
18          205          83
30          32          248
221          126          67
221          182          70
230          2          32
5          205          254
18          24          105
221          126          71
183          0          0
221          229          225
17          82          0
25          17          10
0          6          0
221          78          13
121          183          40
54          229          221
229          197          6
10          126          221
190          71          32
5          35          221
35          16          245
193          221          225
225          40          57
48          4          25
13          24          223
229          213          197
229          203          33
96          105          9
9          9          9
68          77          225
43          9          235
25          235          237
184          193          209
225          221          229
6          10          221
126          71          119
221          35          35
16          247          221
225          221          126
13          60          221
119          13          254
50          40          6
205          18          19
194          109          253
221          229          175
205          247          23
221          229          225
17          44          0
25          205          80
29          62          13
205          102          29
221          229          225
17          82          0
25          221          70
13          120          183
40          5          205
80          29          16
251          205          56
29          123          203
63          215          40
45          62          13
205          102          29
215          227          45
62          13          205
102          29          221
225          205          196
16          201          221
110          26          221
102          27          30
0          14          32
126          35          6
8          31          56
1          28          16
250          13          32
243          201          197
6          10          126
205          102          29
35          16          249
62          13          205
```

```
102   29   229
215   77   13
225   193  201
221   229  215
16    0    221
225   201  205
232   15   221
126   25   205
247   23   219
239   230  1
32    2    231
14    221  229
225   17   82
0     25   229
209   19   1
31    0    175
119   237  176
62    255  221
119   13   1
251   4    237
67    201  92
205   18   19
40    87   205
196   18   205
83    30   32
49    221  126
67    221  182
70    230  2
32    5    205
254   18   24
34    221  229
225   17   71
0     25   1
10    0    205
30    19   32
19    205  6
19    120  182
119   221  203
67    78   40
7     221  126
68    60   221
119   13   221
229   225  17
82    0    25
30    0    14
32    126  35
6     8    31
48    1    28
16    250  13
32    243  221
126   13   187
32    164  205
62    30   205
196   18   205
6     19   40
35    229  197
62    230  211
239   1    104
1     205  250
24    221  229
```

```
225   17   55
0     25   205
120   24   62
238   211  239
205   254  18
193   225  120
47    166  119
221   229  225
17    82   0
25    6    32
126   183  32
200   35   16
249   175  205
247   23   205
196   16   201
175   221  119
67    221  119
69    221  119
70    221  229
225   17   67
0     25   205
65    19   201
221   229  225
17    67   0
25    205  163
24    205  65
19    192  221
203   67   70
201   205  110
29    24   25
205   241  19
24    20   205
88    28   24
15    205  110
27    24   10
205   240  26
24    5    205
218   20   24
0     195  193
5     245  31
31    31   31
205   144  30
241   230  15
254   10   56
2     198  7
198   48   205
169   30   201
245   205  135
30    62   32
205   169  30
241   201  229
213   197  245
217   229  213
197   245  42
81    92   229
245   62   2
215   1    22
241   215  16
0     225  34
81    92   241
```

```
193      209      225
217      241      193
209      225      201
229      197      245
6      10      126
205      158      30
35      16      249
62      13      205
169      30      241
193      225      201
62      14      50
216      92      33
87      253      34
237      92      217
229      217      207
50      217      225
217      251      201
255      255      255
255      255      255
255      255      13
```

Grave o codigo introduzido com a instrucao:

```
SAVE *"m";1;" CAT "CODE 64855,
680
```

Finalmente,verifique com a instrucao:

```
VERIFY *"m";1;" CAT "CODE
```

E tudo.Desligue o computador, torne a liga-lo e tecle:

RUN

e ENTER.O programa entrara automaticamente e fornecera instrucoes de uso.

O programa tem 3 opcoes:

1-Leitura de cabecalhos
2-Catalogo de cartridges
3-Copia de programas

A 1a. opcao permite fazer a leitura dos cabecalhos dos programas,que contem varias informacoes uteis,como o tipo de programas (programa Basic,bytes, ficheiros,etc.),o endereco para onde vai,o comprimento,etc.

A 2a. opcao permite fazer o catalogo de programas que estao na microcassete,incluindo os "invisiveis",cujo nome comeca com o caracter n.0 (representados no catalogo por "?".)

A 3a. opcao permite fazer a copia de programas gravados,de uma cartridge para outra. (Nota:Quando nesta opcao,o programa a copiar leva muito tempo a entrar p/a memoria.Nao se preocupe com a demora.)

SOBRE O PROGRAMA GRAF-3D

«Como se consegue introduzir a 1.ª linha do programa GRAF-3D, editado na revista n.º 24?»

Mário Sancho/Porto

— Basta escrever 1 REM e três linhas de pontos, o que significa apenas uma reserva de espaço para o código máquina que está na linha 130. Quando fizer RUN, observará que a primeira linha fica com os caracteres correspondentes aos que observou na listagem.

PATINHAS

SPECTRUM

António Pereira/Sever do Vouga

(Continuação do n.º anterior)

```
5370 IF IN 2=239 THEN LET y=y-8:
 LET x=x-8: POKE 60005,15: RANDO
MIZE USR 60011: POKE 60005,1: PO
KE 60001,y: POKE 60000,x: LET w=
w-1: RANDOMIZE USR 60011
5375 GO SUB e
5380 GO TO 5360
5400 LET x=x+3: GO SUB b: GO TO
6000
5450 IF IN 2<>239 THEN GO TO 600
0
5455 PRINT INK s;: LET w=0: POKE
 50005,15: RANDOMIZE USR 60011:
POKE 60005,1: LET x=INT (x/8)*8+
3: POKE 60000,x: RANDOMIZE USR 6
0011
5460 PRINT INK s;: IF IN 2=239 T
HEN LET y=y-8: LET x=x-8: POKE 6
0005,15: RANDOMIZE USR 60011: PO
KE 60005,1: POKE 60001,y: POKE 6
0000,x: LET w=w+1: RANDOMIZE USR
 60011
5465 IF (w=0 OR w=4) THEN GO TO
6000
5470 IF IN 2=247 THEN LET y=y+8:
 LET x=x+8: POKE 60005,15: RANDO
MIZE USR 60011: POKE 60005,5: PO
KE 60000,x: POKE 60001,y: LET w=
w-1: RANDOMIZE USR 60011
5475 GO SUB e
5480 GO TO 5460
5500 PRINT INK s;: POKE 60005,15
: RANDOMIZE USR 60011: LET w=y:
POKE 60005,9: LET x=INT (x/8)*8+
1: POKE 60000,x
5510 LET y=y-2: POKE 60001,y: RA
NDOMIZE USR 60011
5520 IF w-y=16 THEN GO TO 1500
5530 GO SUB e: GO TO 5510
5600 LET x=8*INT (x/8)+1: POKE 6
0005,15: RANDOMIZE USR 60011: PO
KE 60000,x
5650 RESTORE 9010: FOR f=1 TO 31
: PRINT INK s;: READ i: POKE 600
```

```
05,15: RANDOMIZE USR 60011: POKE
 60005,9: POKE 60001,i: RANDOMIZ
E USR 60011: GO SUB e: NEXT f
5660 IF ABS (k-x)<10 THEN LET g=
e: LET e=420: LET b=420: GO SUB
380
5665 IF a(15,30)=5000 THEN LET a
(15,30)=0: GO SUB 5080
5670 RESTORE 9012: FOR f=1 TO 5:
 PRINT INK s;: READ i: POKE 6000
1,i: RANDOMIZE USR 60011: GO SUB
 e: NEXT f
5680 PRINT INK s;: LET x=x-5: PO
KE 60005,15: RANDOMIZE USR 60011
: POKE 60005,1: POKE 60000,x: RA
NDOMIZE USR 60011: GO TO 6000
5700 LET x=8*INT (x/8)+1: POKE 6
0005,15: RANDOMIZE USR 60011: PO
KE 60000,x
5750 RESTORE 9020: FOR f=1 TO 14
: PRINT INK s;: READ i: POKE 600
05,15: RANDOMIZE USR 60011: POKE
 60005,9: POKE 60001,y-i: RANDOM
IZE USR 60011: GO SUB e: NEXT f
5760 RESTORE 9011: FOR f=1 TO 17
: PRINT INK s;: READ i: POKE 600
05,15: RANDOMIZE USR 60011: POKE
 60005,9: POKE 60001,i: RANDOMIZ
E USR 60011: GO SUB e: NEXT f
5770 PRINT INK s;: LET x=x-5: PO
KE 60005,15: RANDOMIZE USR 60011
: POKE 60005,1: POKE 60000,x: RA
NDOMIZE USR 60011
5780 LET y=119: GO TO 6000
5800 LET x=8*INT (x/8)+1: POKE 6
0005,15: RANDOMIZE USR 60011: PO
KE 60000,x
5850 RESTORE 9010: FOR f=1 TO 19
: PRINT INK s;: READ i: POKE 600
05,15: RANDOMIZE USR 60011: POKE
 60005,9: POKE 60001,i: RANDOMIZ
E USR 60011: GO SUB e: NEXT f
5890 LET y=i: POKE 60001,y: GO T
O 6000
5900 BEEP .005,10*COS x
5925 IF IN 2<>254 THEN GO TO 600
0
5930 FOR f=1 TO 5: BEEP .01,f: G
O SUB e: NEXT f: FOR f=5 TO 1 ST
EP -1: BEEP .01,f: GO SUB e: NEX
T f
5935 POKE 23675,84: POKE 23676,2
36: PRINT AT 19-INT (y/8),INT (x
/8);" ";AT 23-INT (y/8),INT (x/
8);"HI": POKE 23675,50: POKE 236
76,235
5940 GO TO 6000
5950 IF IN 2<>247 THEN GO TO 600
0
5955 PRINT INK s;: LET w=y: POKE
 60005,15: RANDOMIZE USR 60011:
POKE 60005,9: POKE 60003,24: LET
 x=INT (x/8)*8+1: POKE 60000,x:
RANDOMIZE USR 60011
5960 PRINT INK s;: IF IN 2=247 T
HEN LET y=y+2: POKE 60001,y: RAN
DOMIZE USR 60011
5965 IF (w-y=0 OR y-w=32) THEN B
EEP .005,5: POKE 60003,16: GO TO
 6000
5970 IF IN 2=239 THEN LET y=y-2:
 POKE 60001,y: RANDOMIZE USR 600
11
5975 GO SUB e
5980 GO TO 5960
6000 GO SUB b
6010 GO SUB 10*IN 2
6020 LET z=a(INT (y/8),1+INT (x/
8))
6030 IF z>0 THEN GO TO z+50*POIN
T (x+4,y-16)
6040 GO TO 6000
7540 REM
============   CODE   ===========
=====  Linhas 7540 TO 7750  ====
=====  devem ser apagadas   ====
=====  depois de gravadas   ====
================================
7550 POKE 61207,71: POKE 61208,2
38: POKE 61013,23: POKE 61014,23
9
7554 POKE 61205,25: POKE 61206,2
39: POKE 61223,21: POKE 61224,23
9
7600 RESTORE 7602: FOR f=60800 T
O 60911: READ u: POKE f,u: NEXT
f
7602 DATA 8,151,24,143,120,147,2
16,143,232,151
7604 DATA 128,175,112,175,104,16
7,112,159,128,159,136,151,128,14
3,112,143
7606 DATA 13,11,13,9,4,9,1,1,13,
11,13,9,4,9,1,1,13,15,16,15,16,1
3,15,13,15,11,13,11,13,15,16,16
7610 DATA 62,64,1,0,0,95,128,87,
123,89,4,203,88,40,10,111,121,19
8,32,79,125,32,2,198,8,203,152,1
11,128,103,125,105,197,1,32,0,23
7,176,193,254,88,32,218
7611 DATA 201,6,88,197,205,186,2
37,193,16,249,201
7690 RESTORE 7695: FOR f=60210 T
O 60464 STEP 2: READ u,v: POKE f
,u: POKE f+1,v: NEXT f
7695 DATA 30,0,30,0,55,0,30,0,62
,0,30,0,12,0,21,0,42,128,85,64,1
70,192,222,0,62,0,183,0,225,128,
99,128
7700 DATA 3,192,3,192,7,96,3,192
,3,224,3,192,1,128,5,64,10,160,2
1,80,26,168,3,216,3,224,7,104,12
,56,14,48
7705 DATA 0,0,0,0,15,128,15,128,
63,224,10,128,15,128,13,128,15,1
28,7,0,21,64,106,176,213,48,31,1
28,57,192,48,192,255,248,170,168
,85,80,42,160,21,64,10,128,0,0,0
,0
7710 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0
7715 DATA 30,0,255,0,47,128,123,
128,127,128,63,0,30,0,15,0,13,12
8,30,192,55,192,239,192,223,192,
25,128,49,128,115,128
7720 DATA 3,192,7,248,15,160,14,
240,15,240,7,224,3,192,7,128,13,
128,27,192,31,96,31,184,31,216,2
8,192,12,96,14,112
7722 DATA 62,252,56,60,54,124,48
,28,62,108,48,12,62,124,0,0
7724 DATA 0,0,15,128,15,128,63,2
24,10,128,15,128,13,128,15,128,7
,0,21,64,42,160,53,96,7,0,15,128
,24,192,0,0
7730 RESTORE 7732: FOR f=60500 T
O 60603: READ u: POKE f,u: NEXT
f
7732 DATA 16,48,127,255,127,48,1
6,0
7734 DATA 8,12,254,255,254,12,8,
0
7736 DATA 255,213,171,213,171,25
5,0,0
7738 DATA 3,14,58,98,79,218,146,
154,192,112,92,70,242,67,65,241,
143,130,194,79,98,58,14,3,89,73,
91,242,70,92,112,192
7740 DATA 127,85,42,21,10,5,0,0,
252,84,168,80,160,64,0,0
7742 DATA 62,56,54,48,62,48,62,0
,252,60,124,28,108,12,124,0
7744 DATA 63,0,1,63,96,255,0,0,2
52,8,248,0,0,255,0,0
7750 SAVE "pt bytes"CODE 60000,1
415
7753 REM                      ==
========  Inicio do 1º quadro  ==
========                      ==
========  Variaveis e grafismo ==
================================

7754 LET p=0: LET c=208: LET q=8
00
7755 LET qqq=q: LET s=2: PAPER 6
: BORDER 6: CLS : BEEP .001,50:
LET h=1: DIM a(22,32): RESTORE 7
```

```
756: FOR x=1 TO 23: READ u,v,z:
LET a(u,v)=z: NEXT x
7756 DATA 10,1,4000,14,1,4200,14
,31,4600,2,1,4000,6,1,4200,10,31
,4100,6,31,4100,2,31,4600,10,2,4
500,6,2,4500,6,16,4500,10,30,450
0,6,30,4500
7757 DATA 14,9,4700,14,23,4700,1
4,21,4400,2,2,5900,2,16,4300,2,3
0,4300,14,2,4400,14,11,4400,10,1
6,4400,14,30,4400
7760 POKE 23675,84: POKE 23676,2
36: RESTORE 7760: FOR f=1 TO 3:
READ u: PRINT INK 5;AT 0,u;"████
██"; INK 7;AT 1,u;"█    █";AT 2,
u;"█    █";AT 3,u;"█    █";AT 4,
u;"█    █": PRINT INK 3;AT 1,u+2
;"DE";AT 2,u+2;"FG": NEXT f: DAT
A 0,13,26
7762 INK 1: PRINT AT 5,0;"CCCCCC
CCCCCCCCCCCCCCCCCCCCCCCC";AT 9
,0;"C  CCCCCCC              CCCCCC
C  C"
7764 PRINT AT 13,0;"C  CCCCCCCCC
CCC  CCCCCCCCCCC  C";AT 17,0;"C
  CCCCCCCCCCC  CCCCCCCCCCC  C"
;AT 21,0;"C  CCCCCCCCCCC  CCCCC
CCCCCCC  C"
7765 PRINT INK 2;AT 6,8;"JK";AT
6,22;"JK"
7766 PRINT INK 3;AT 6,0;"A";AT 6
,31;"B";AT 14,0;"A";AT 18,31;"B"
7768 PRINT INK 2;AT 9,1;"HI";AT
13,15;"HI";AT 17,29;"HI";#0; INK
 2;TAB 1;"HI"
7770 PRINT INK 2;: POKE 23675,50
: POKE 23676,235: RESTORE 7770:
POKE 60002,13: POKE 60003,16: PO
KE 60001,175: FOR f=1 TO c/16-10
: READ u,v: POKE 60000,u: POKE 6
0005,v: RANDOMIZE USR 60011: NEX
T f: DATA 160,5,176,9,192,1
7771 LET aa=0: LET r=60825: LET
e=410: LET b=400: LET a=60799: L
ET m=0: LET x=48: LET y=119: LET
 u=200: LET v=87: LET w=0
7772 PRINT INK s;: POKE 60005,1:
 POKE 60000,x: POKE 60001,y: RAN
DOMIZE USR 60011
7773 POKE 61002,13: POKE 61003,1
6: POKE 61000,u: POKE 61001,v: P
OKE 61005,19: RANDOMIZE USR 6101
1
7774 LET d=700: LET k=8: LET l=2
3: POKE 61210,k: POKE 61211,l: P
OKE 61212,16: POKE 61213,7: POKE
 61215,27: RANDOMIZE USR 61221
7776 PRINT INK 0;AT 3,19;"QUADRO
";AT 4,20;"123";AT 4,19+h; FLASH
 1;h: PRINT FLASH 0
7778 PRINT AT 0,7;"SCORE";AT 1,7
;"000";AT 3,7;"BONUS";AT 4,7;q;A
T 1,7;p
7779 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
7780 RETURN
7782 REM                       ==
=======  Inicio do ■´ quadro  ==
=======                       ==
=======  Variaveis e grafismo ==
================================

7785 LET p=0: LET c=208
7790 LET s=5: BORDER 0: PAPER 0:
 CLS : BEEP .001,50: DIM a(22,32
): LET h=2
7794 RESTORE 7795: FOR x=1 TO 52
: READ u,v,z: LET a(u,v)=z: NEXT
 x
7795 DATA 14,4,4800,10,4,4800,2,
4,4800,10,15,4800,6,15,4800,14,6
,4900,10,6,4900,2,6,4900,10,17,4
900,6,17,4900,10,2,5000,6,3,5000
,2,18,5000
7796 DATA 10,8,4800,10,9,4800,10
,18,4800,10,19,4800,10,12,4900,1
0,13,4900,10,22,4900,10,23,4900,
10,10,5500,10,11,5500,10,20,5500
,10,21,5500
```

```
7797 DATA 2,8,4100,10,25,5100,6,
21,5100,6,29,5200,2,25,5200,2,7,
5300,6,11,5400,14,5,5600,6,31,46
00,10,1,4000,6,1,4000,2,1,4000,2
,31,4100
7799 DATA 2,24,4000,6,28,4000,2,
12,4000,2,19,4100,6,30,4400,6,16
,4500,10,16,4500,14,17,4700,2,16
,5900,2,30,4300,14,16,4400,14,18
,4400,10,5,4400,6,5,4400
7810 INK 3: POKE 23675,84: POKE
23676,236: PRINT AT 5,0;"CCCCCCC
CCCCCCC  CCCCCCCCCCCCCC";AT 9,
0;"CCCC  CCCCCCCCC";AT 13,0;"CCC
C  CCC  CCC  CC  CCC"
7813 PRINT AT 17,0;"CCCC       C
CCC  CCCC    CCC    C";AT 21,0;"C
CCC  CCCCCCCCC  CCCCCCCCCCC  C"
7814 PRINT AT 14,24;"CC";AT 15,2
4;"CCC";AT 16,24;"CCCC";AT 17,25
;"CCCC";AT 18,10;"CC          CC";
AT 19,9;"CCC          CCC";AT 20,8
;"CCCC        CCCC"
7816 PRINT INK 4;AT 12,1;"JK";AT
 16,2;"JK";AT 20,17;"JK"; INK 2;
AT 5,15;"JK"
7818 INK 5: PRINT AT 21,4;"LM";A
T 13,15;"HI";AT 17,29;"HI";#0; I
NK 5;TAB 15;"HI"
7819 PRINT INK 7;AT 14,31;"B"
7820 RESTORE 7820: FOR f=1 TO 3:
 READ u: PRINT INK 2;AT 0,u;"███
███"; INK 6;AT 1,u;"█    █";AT 2
,u;"█    █";AT 3,u;"█    █";AT 4
,u;"█    █": PRINT INK 7;AT 1,u+
2;"DE";AT 2,u+2;"FG": NEXT f: DA
TA 0,13,26
7824 LET aa=0: LET q=400: LET r=
60825: LET g=800: LET e=800: LET
 k=8: LET l=127: LET b=800: LET
a=60799: LET m=0: LET x=144: LET
 y=87: LET w=0
7825 INK 5: POKE 23675,50: POKE
23676,235: RESTORE 7822: POKE 60
002,13: POKE 60003,16: POKE 6000
1,175: FOR f=1 TO c/16-10: READ
u,v: POKE 60000,u: POKE 60005,v:
 RANDOMIZE USR 60011: NEXT f: DA
TA 160,5,176,9,192,1
7826 POKE 61210,k: POKE 61211,l:
 POKE 61212,16: POKE 61213,7: PO
KE 61215,27: RANDOMIZE USR 61221
7828 POKE 60005,1: POKE 60000,x:
 POKE 60001,y: RANDOMIZE USR 600
11
7830 INK 7: PRINT AT 3,19;"QUADR
O";AT 4,20;"123";AT 4,19+h; FLAS
H 1;h: PRINT FLASH 0
7832 PRINT AT 3,7;"BONUS";AT 4,7
;q;AT 0,7;"SCORE";AT 1,7;"000";A
T 1,7;p
7839 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
7840 RETURN
7845 REM                       ==
=======  Inicio do ■´ quadro  ==
=======                       ==
=======  Variaveis e grafismo ==
================================

7850 LET qqq=500: LET p=0: LET c
=208
7855 LET s=1: PAPER 7: BORDER 7:
 CLS : BEEP .001,50: LET h=3: DI
M a(22,32)
7860 RESTORE 7861: FOR x=1 TO 54
: READ u,v,z: LET a(u,v)=z: NEXT
 x
7861 DATA 18,29,5000,10,16,5900,
18,1,4200,14,5,4000,2,8,4000,6,1
5,4000,2,4,4100,6,11,4100,6,23,4
100,10,18,4100,10,1,4200,2,1,420
0,14,16,4400,18,26,4400,6,19,440
0,10,29,4600,10,31,4600,2,31,460
0
7862 DATA 14,29,5000,14,18,5500,
18,31,4600,14,30,5800,18,27,4700
,10,11,5500,10,8,5000,10,10,5000
,10,28,5000,2,13,5000,2,15,5000,
```

```
6,16,5000
7863 DATA 18,2,5100,2,19,5900,14
,6,5200,2,3,5300,6,10,5300,6,22,
5300,10,17,5300,14,21,5400,10,14
,5400,10,26,5400,6,7,5400,10,30,
5700,18,27,4700
7864 DATA 14,8,4800,14,9,4800,10
,2,4800,10,3,4800,14,11,4900,14,
12,4900,10,6,4900,10,7,4900,14,1
0,4400,10,4,5500,10,5,5500
7870 POKE 23675,84: POKE 23676,2
36: PRINT INK 3; PAPER 7;AT 5,0;
"CC";AT 5,27;"CCCCC";AT 9,0;"CCC
CCCCCC";AT 9,11;"CCCC";AT 9,21;"
CCCCCCCC";AT 13,0;"CCC";AT 13,6;
"CCCC";AT 13,14;"C  CCCCC";AT 13
,26;"CCC";AT 13,31;"C"
7872 PRINT INK 3; PAPER 7;AT 17,
7;"CCCCCCCCCC  CCCCCCCCC";AT 21
,0;"CCCCCCCCCCCCCCCCC  CCCCCCCC
C  C"
7874 PRINT PAPER 9; INK 2;AT 2,2
7;"JK"; INK 4;AT 4,28;"JK";AT 8,
28;"JK";AT 12,7;"JKJK";AT 12,27;
"JK";AT 16,15;"JK";AT 20,12;"JKJ
K"
7876 PRINT INK 1;AT 17,18;"HI";A
T 13,15;"HI";AT 14,15;"HI";#0; I
NK 1;TAB 18;"HI"
7878 PRINT INK 0;AT 2,0;"A";AT 2
,31;"B";AT 10,0;"A";AT 10,28;"B
B";AT 18,0;"A";AT 18,31;"B"
7880 RESTORE 7880: FOR f=1 TO 4:
 READ u,v: PRINT INK 3; PAPER 7;
AT u,v;"CC";AT u+1,v-1;"CCC";AT
u+2,v-2;"CCCC": NEXT f: DATA 10,
20,14,13,14,25,18,6
7882 PRINT INK 3; PAPER 7;AT 6,1
;"CC";AT 7,1;"CCC";AT 8,1;"CCCC"
; INK 1;AT 21,29;"LM"
7892 LET aa=0: LET q=500: LET r=
60825: LET e=900: LET b=900: LET
 a=60809: LET m=0: LET x=96: LET
 y=119: LET w=0: LET v=119: LET
u=160: LET j=-3
7894 INK 1: POKE 23675,50: POKE
23676,235: RESTORE 7822: POKE 60
002,13: POKE 60003,16: POKE 6000
1,175: FOR f=1 TO c/16-10: READ
k,l: POKE 60000,k: POKE 60005,l:
 RANDOMIZE USR 60011: NEXT f: DA
TA 160,5,176,9,192,1
7896 PRINT INK 1;: POKE 60005,1:
 POKE 60000,x: POKE 60001,y: POK
E 60002,13: POKE 60003,16: RANDO
MIZE USR 60011
7898 INK 0: PRINT AT 3,19;"QUADR
O";AT 4,20;"123";AT 4,19+h; FLAS
H 1;h: PRINT FLASH 0
7900 PRINT AT 3,7;"BONUS";AT 4,7
;q;AT 0,7;"SCORE";AT 1,7;"000";A
T 1,7;p
7905 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
7910 POKE 61001,v: POKE 61002,13
: POKE 61003,16: POKE 61005,19
7919 RETURN
9000 DATA 2,1,4,2,6,3,8,4,10,4,1
2,4,14,4,16,3,18,2,20,1,22,0
9010 DATA 117,114,111,107,103,99
,94,89,84,79,73,67,61,55,49,42,3
5,28,23
9011 DATA 31,39,47,55,63,72,81,9
0,99,108,118,126
9012 DATA 125,124,122,120,119
9020 DATA 2,5,8,12,16,20,25,30,3
5,40,46,52,58,64
9600 REM
========== INSTRUCOES ==========

9610 BORDER 4: PAPER 4: INK 4: C
LS
9620 FOR f=23232 TO 23295: POKE
f,8*4+4: NEXT f
9630 FOR f=0 TO 10: PRINT AT f,3
; INK 1; PAPER 5;"
            ": NEXT f
9650 PRINT AT 11,3;"...... INSTR
UCOES: ......"'"    PATINHAS TE
RA' DE LEVAR"'"   P/ A SUA CAIXA
 FORTE"'"   TODAS AS NOTAS DO QU
ADRO."'"    DEVERA' ENTREGA'-LAS
 POR"'"    BAIXO DA NOTA VERMELHA
."'"    SO' PODE APANHAR UMA"'"
  NOTA DE CADA VEZ."'"      FICARA
  DA SUA COR"'"   QUANDO O FIZER
 "
9651 RANDOMIZE USR 60902
9660 PRINT AT 11,4;"SE TOCAR NOU
TRA NOTA SEM"'"   ARRECADAR A AN
TERIOR"'"   MORRERA'.
       "'"

           DEVE EVITAR O METRAL
HA    "'"     E OS PRECIPICIOS.
      "
9661 FOR F=17 TO 21: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9662 GO SUB 9800
9665 RANDOMIZE USR 60902
9670 PRINT AT 11,3;" O JOGO CONS
TA DE          "'"    3 QUADROS DIS
TINTOS.       "'"     AO COMPLETAR
O QUADRO      "'"    ANTERIOR PASSA
DE          "'"    IMEDIATO AO SEGU
INTE.        "
9671 FOR F=16 TO 18: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9672 PRINT AT 19,3;" QUANTO MAIS
 RAPIDO        "'"    TERMINAR UM Q
UADRO, MAIOR"'"    SERA' O BONUS.
          "
9673 GO SUB 9800
9675 RANDOMIZE USR 60902
9680 PRINT AT 11,3;" SE DEIXAR O
 BONUS CHEGAR"'"    A 0000, MORRE
RA'.          "'"
              "'"     O BONUS DI
MINUI C/ A     "'"    PASSAGEM DO
3' P/ O 1'    "'"    QUADRO.
          "
9681 FOR F=17 TO 19: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9682 PRINT AT 20,3;" CADA QUADRO
 TEM UM BONUS"'"    DIFERENTE.
          "
9683 GO SUB 9800
9685 RANDOMIZE USR 60902
9690 FOR F=11 TO 21: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9693 GO SUB 9800
9695 RANDOMIZE USR 60902
9700 PRINT AT 11,3;"....... COMA
NDOS: ......."'"
             "'"    R ...........
...... SUBIR"'"    G ............
.... DESCER"'"    K .............
. ESQUERDA"'"    L ..............
. DIREITA"
9701 PRINT AT 17,3;"R+K .... SAL
TAR P/ A ESQ."'"    R+L .... SALT
AR P/ A DIR."'"    ENTER . CHAMAR
 O ELEVADOR"
9702 PRINT AT 20,3;"K+L ........
....... PAUSA"'"    G+K+L .. ABAN
DONAR O JOGO"
9703 GO SUB 9800
9705 RANDOMIZE USR 60902
9710 PRINT INK 3; PAPER 7;AT 16,
8;"PRIMA UMA TECLA";AT 17,8;" PA
RA COMECAR. ";AT 18,8;"
 "
9711 BEEP .01,10*RND
9712 IF INKEY$="" THEN GO TO 971
1
9720 FOR F=11 TO 14: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9721 PRINT AT 15,3;" ESCOLHA O Q
UADRO ONDE     "'"    DESEJA JOGAR
: 1,2 OU 3 "
9722 FOR F=17 TO 21: FOR G=3 TO
27: PRINT AT F,G;" ": NEXT G: NE
XT F
9725 RANDOMIZE USR 60902
```

```
9730 RETURN
9800 FOR F=10 TO 60: BEEP .001,F
 : NEXT F
9810 FOR F=60 TO 10 STEP -1: BEE
P .001,F: NEXT F
9820 RETURN
9998 STOP
9999 CLEAR : SAVE "pt program" L
INE 40

9400 REM
===========  SCREEN$  ===========

9500 PAPER 0: BORDER 0: FLASH 0:
 OVER 0: INVERSE 0: BRIGHT 0: CL
S : PRINT AT 9,1; PAPER 6; INK 2
;"SOFTW."
9505 INK 5: FOR f=0 TO 3: PLOT 1
1,105+f: DRAW 0,-11-2*f,PI: DRAW
 41,0: DRAW 0,11+2*f,PI: DRAW IN
K 3;-41,0: NEXT f
9510 PRINT PAPER 6; INK 2;AT 9,2
5;"BASTOS"
9515 FOR f=0 TO 3: PLOT 203,105+
f: DRAW 0,-11-2*f,PI: DRAW INK 3
;41,0: DRAW 0,11+2*f,PI: DRAW -4
1,0: NEXT f: PLOT INK 3;243,93
9520 PRINT INK 6; PAPER 2;AT 20,
3;" ESPERE "; INK 2; PAPER 6;AT
20,21;"PROGRAMA";AT 21,3;"UM POU
CO"; INK 6; PAPER 2;AT 21,21;"A
ENTRAR"
9525 INK 1: FOR f=9 TO 20: PRINT
 AT f,14;"█  █": NEXT f
9530 FOR f=48 TO 55: CIRCLE 128,
55,f: NEXT f
9535 INK 3: FOR f=12 TO 19: CIRC
LE 112,68,f: CIRCLE 142,44,f: NE
XT f
9540 FOR f=11 TO 18: PRINT AT f,
15;"  ": NEXT f
9545 FOR f=49 TO 56: PLOT 120,f:
 DRAW 16,6: NEXT f
9550 PRINT AT 11,15;"████";AT 18
,13;"████"
9560 INK 1: FOR f=1 TO 3: PRINT
AT 11+f,14;"█";AT 14+f,17;"█": N
EXT f
9570 PRINT INK 4;AT 0,0;"████████
████████████████████████████████"
9580 INK 1: LET b=0: LET a=0: FO
R f=0 TO 56 STEP 2: PLOT 0,f: DR
AW 73+b,70-a: PLOT 255,f: DRAW -
73-b,70-a: LET a=a+.78: LET b=b+
1.1: NEXT f
9590 SAVE "ptSCREEN$ "SCREEN$
```

NOTAS SOBRE O PROGRAMA «PATINHAS»

O «coração do programa encontra-se nas linhas 6000 até 6040:

— 6000 — Dá-nos a possibilidade de variar os possíveis movimentos de outras figuras no écran, além do Patinhas. Para isso basta alterar o valor da variável «b» consoantte convier.

— 6010 — Faz a leitura do teclado.

— 6020 e 6030 — Vão usar o dimensionamento do écran para que se faça o movimento do Patinhas nos termos aí estabelecidos previamente.

— 6040 — Retoma o ciclo.

Consulte nas listagens as Linhas REM que o elucidarão um pouco mais.

IMPORTANTE! Quase todas as linhas do programa estão numeradas umas em função de outras. Por isso nunca altere o número das linhas; caso contrário o programa poderá não funcionar.

CONSTRUÇÃO DO PROGRAMA

«PATINHAS», no final, será constituído por 4 partes:

— 1.ª parte — pt entrada
— 2.ª parte — pt screen$
— 3.ª parte — pt bytes
— 4.ª parte — pt program

Grave-as por esta ordem! Os processos de gravação estão sempre na última linha da parte em causa.

ENTRADA

— Pt Entrada — Linhas 1 e 2 — Basta fazer RUN2
— Pt Screen$ — Linhas 9400 a 9590 — RUN 9500
— Pt Bytes — Linhas 10, 20 e de 7540 até 7750

- 1.º passo: Consulte o n.º 19, págs. 11 a 15 deste jornal e introduza os bytes «animate». Grave os bytes. Experimente RANDOMIZE USR 65 171. OK?
- 2.º passo: Entre as linhas indicadas no início.
- 3.º passo: Faça GOTO 10. Como poderá ver esta linha pede-lhe 3 cópias dos bytes do 1.º passo. Ande com a fita para onde os gravou e coloque o gravador na posição de ler. Repita isto 3 vezes. Deixe o programa correr. A linha 7750 gravará estes bytes. Pode agora destruir as linhas 10,20 e de 7540 a 7750. É, no entanto, aconselhável gravá-las em BASIC para melhor detecção de eventuais erros.

— Pt Programa — Linhas 1 a 9999 — Devido à sua extensão, o melhor será introduzi-las por partes. Depois reúna-as com MERGEs.

Antes de experimentar qualquer parte deste programa, será melhor gravá-la primeiro.
Lembre-se que um RANDOMIZE USR errado pode levar à destruição do programa.
GOTO 9999 grava «pt program».

DIVIRTA-SE!

NOTA: O programa foi várias vezes experimentado e tudo correu bem. No entanto, se tiver problemas ou quiser esclarecer alguma dúvida, encontro-me à disposição dos sócios.

SCROLL NO SPECTRUM

«Na revista de Setembro pág. 5, Carlos Moreno apresenta uma subrotina para efectuar SCROLL no Spectrum.

Acrescento que o Spectrum já possui SCROLL na ROM: RAND USR 3582».

Hugo Assumpção
Lisboa

JOGO DA VIDA

SPECTRUM

ESTE É O MAIS CURTO E RÁPIDO JOGO DA VIDA, QUE JÁ VIMOS.

As suas regras são muito simples:

1) Uma célula nasce, se um espaço vazio tem exactamente três células «vizinhas».
2) Uma célula morre, se possui mais do que três ou menos do que dois «vizinhos».

Um écran aleatório é gerado de modo a apresentar uma colónia de células, e sucessivamente, vai apresentando outras colónias.

Eventualmente uma colónia morre, tornando-se estável e estática ou estável e dinâmica.

Pode premir uma tecla qualquer para regressar ao BASIC. Para recomeçar o programa, substitua as linhas 10 e 20 por

10RANDOMIZE USR 32768
20 GOTO 10
RUN

O PROGRAMA PRODUZ CERCA DE 9 GERAÇÕES POR SEGUNDO.

O CÓDIGO — cerca de 159 bytes — é Simples de «desassemblar» e modificar.

```
Jogo da Vida.....LIFE Game

   5 REM jogo da 'VIDA'
  10 CLEAR 32767
  20 FOR f=32768 TO 32926
  30 READ n: POKE f,n: NEXT f
  40 BORDER 0: PAPER 0: CLS
  50 RANDOMIZE USR 32768: GO TO
50
 100 DATA 4,17,6,3,33,-2,-1,58,2
52,-1,31,31,31,31,174,15,203,22,
43,203,22,43
 110 DATA 203,22,43,203,22,58,-2
,-1,230,1,237,68,2,3,27,122,179,
32,219,221,33,0
 120 DATA 88,33,0,129,17,0,88,1,
0,3,237,176,219,-2,238,-1,192,38
,88,17,0,91
 130 DATA 14,32,237,176,33,224,9
0,17,224,87,14,32,237,176,38,129
,1,0,3,221,126,1
 140 DATA 237,68,221,86,33,146,2
21,86,32,146,221,86,31,146,221,8
6,-1,146,221,86,223,146
 150 DATA 221,86,224,146,221,86,
225,146,221,86,0,245,122,167,40,
23,241,54,0,-2,4,48
 160 DATA 6,-2,2,56,2,54,-1,35,2
21,35,11,120,177,32,194,24,148,1
19,241,254,3,32
 170 DATA 240,54,255,24,236
```

PROGRAMAS PARA RÁDIO-AMADORES

KEN WILLIAMS pergunta:
USA EFECTIVAMENTE O SEU MICRO???

Algum tempo atrás, um amigo ofereceu-me uma caixa com algumas dezenas de Cristais, e eu pensava que poderia aproveitar um ou outro para os «dois metros».

Liguei o meu microcomputador, liguei o TV, coloquei uma cassette no gravador, e eis-me atirado ao trabalho, usando efectivamente o microcomputador.

10 REM programa para verificar a frequência de um cristal.

Aqui está o início de um programa, assinalando a finalidade do programa; agora vamos ao cálculo:

primeiro a frequência do cristal:

20 PRINT «seleccionar a freq. do cristal»
30 INPUT f

Agora para dois metros, podemos usar X3, X8, X9, X12, X18, X24, de modo que teremos de escrever algumas linhas mais.

40 PRINT "3F = " ; 3*3F
50 PRINT "8F = " ; 8*F
60 PRINT "9F = " ; 9*F

ETC. Estas linhas estão relacionadas com as frequências de transmissão, todavia vamos supor que o CRISTAL se destina a um RECEPTOR?

Felizmente o meu equipamento usa uma IF de 10.7 Mhz, portanto o programa deve ser lido como:

100 PRINT "3F + 10.7 = " ; (3*F) + 10.7
etc.

tendo o devido cuidado com as regras da matemática, deixemos sempre ao computador o efectivar dos cálculos, de modo a calcular de seguida as possibilidades do próximo cristal, e isso requere:

160 PRINT «próxima frequência do cristal?»
170 GOTO 30

e voltamos atrás para o início do programa.

É evidente que pude verificar a utilidade de um programa deste tipo, e em 5 minutos tive uma lista das possibilidades de cada cristal, existente na minha caixa.

Com algumas substituições e modificações, eu consegui produzir um programa mais útil, dado que nem todos os receptores usam um cristal de 10.7 Mhz.

O nosso programa vai portanto perguntar a frequência intermédia do receptor (IF).

Inicialmente podemos indicar qual o número de cristais requeridos para as frequências dadas e seleccionamos 1 ou 2. Temos sempre a possibilidade de regressar ao MENU após casa cálculo efectuado ou continuamos a entrar com frequências, etc.

```
Programa de calculo -FREQUENCIAS
 disponiveis para determinados
           CRISTAIS


Exemplo : Freq. do Cristal


10.7

Prox. Freq. do Cristal... 8
```

```
3    F    = 24
8    F    = 64
9    F    = 72
12   F    = 96
18   F    = 144
24   F    = 192
36   F    = 288
3F+IF     = 34.7
8F+IF     = 74.7
9F+IF     = 82.7
12F+IF=  106.7
18F+IF=  154.7
24F+IF=  202.7
36F+IF=  298.7

seleccionar 0(Menu) ou 1 (cont.)
```

```
  10 BORDER 0: PAPER 0: INK 7
  20 CLS
  30 PRINT "Programa de calculo
-FREQUENCIAS disponiveis para de
terminados           CRISTAIS"
  40 PRINT AT 20,0;"Seleccionar
1 ou 2"
  50 PAUSE 0: LET n$=INKEY$
  60 IF n$<"1" OR n$>"2" THEN GO
 TO 40
  70 IF n$="1" THEN GO TO 180
  80 IF n$="2" THEN GO TO 500
 170 REM verificar a freq.do cri
stal
 180 CLS
 190 INPUT "qual a freq. interm.
 do receptor ";if
 200 CLS
 210 PRINT ''
 220 INPUT "Frequencia do CRISTA
L ";f
 250 PRINT "3   F   = ";3*f
 260 PRINT "8   F   = ";8*f
 270 PRINT "9   F   = ";9*f
 280 FOR i=2 TO 6
 285 IF i=5 THEN GO TO 300
 290 PRINT i*6;" F   = ";i*6*f
 300 NEXT i
 310 PRINT "3F+IF = ";3*f+if
 320 PRINT "8F+IF = ";8*f+if
 330 PRINT "9F+IF = ";9*f+if
 340 PRINT "12F+IF= ";12*f+if
 350 PRINT "18F+IF= ";18*f+if
 360 PRINT "24F+IF= ";24*f+if
 370 PRINT "36F+IF= ";36*f+if
 400 PRINT AT 20,0;"seleccionar
 0(Menu) ou 1 (cont.)"
 410 PAUSE 0: LET n$=INKEY$
 420 IF n$<"0" OR n$>"1" THEN GO
 TO 410
 430 IF n$="0" THEN GO TO 20
 440 CLS
 460 PRINT "selec. proxima freq.
 CRISTAL"
 470 GO TO 210
 500 CLS
 510 INPUT "Qual a freq. interm.
do receptor";a
 520 PRINT ''
 530 INPUT "Frequencia requerida
?";f
 550 PRINT
 560 PRINT "Cristal de transmiss
ao "
 570 PRINT "X3   =";f/3
 580 PRINT "X6   =";f/6
 590 PRINT "X8   =";f/8
 600 PRINT "X9   =";f/9
 610 PRINT "X12  =";f/12
 620 PRINT "X18  =";f/18
 630 PRINT "X24  =";f/24
 640 PRINT "X36  =";f/36
 650 PRINT "Cristal do receptor"
 660 PRINT "X3   =";(f-a)/3
 670 PRINT "X6   =";(f-a)/6
 680 PRINT "X8   =";(f-a)/8
 690 PRINT "X9   =";(f-a)/9
 700 PRINT "X12  =";(f-a)/12
 710 PRINT "X18  =";(f-a)/18
 720 PRINT "X24  =";(f-a)/24
 730 PRINT "X36  =";(f-a)/36
 740 PRINT
 750 PRINT AT 20,0;"seleccionar
 0(Menu) ou 1 (cont.)"
 760 PAUSE 0: LET n$=INKEY$
 770 IF n$<"0" OR n$>"1" THEN GO
 TO 770
 780 IF n$="0" THEN GO TO 20
 790 CLS
 800 INPUT "proxima Frequencia n
ecessaria ? ";f
 810 GO TO 550
```

OS 10 MAIS VENDIDOS

EM INGLATERRA	NO CLUBE Z80
1. Combat Lynx *	Pole Position
2. Pyjamarama *	Decathlon
3. Decathlon *	Chuckie Egg
4. Sherlock Holmes	Sabre Wulf
5. Avalon	Full Throttle
6. Beach-Head *	Match Point
7. Kentilla	Pyjamarama
8. Dark Star	Code Name Mat
9. Delta Wing	Psitron
10. Full Throttle *	Hulk

* Programas disponíveis no Clube Z80.

COMPRO OU TROCO

Por uma cassette que já possua (Compiler II, Assembler/Disassembler, Jogos) a CASSETTE BETA BASIC 1.0 para Spectrum 16/48 K.

Contactar:

RUI CARVALHO — R. Miguel Pais n.º 31-3.º D
2830 BARREIRO

THE QUILL

QUEM RESPONDE?

Como traduzir as mensagens «standard» do programa THE QUILL para que, nos jogos, apareçam em português?

Pedro Ferrão/Mirandela

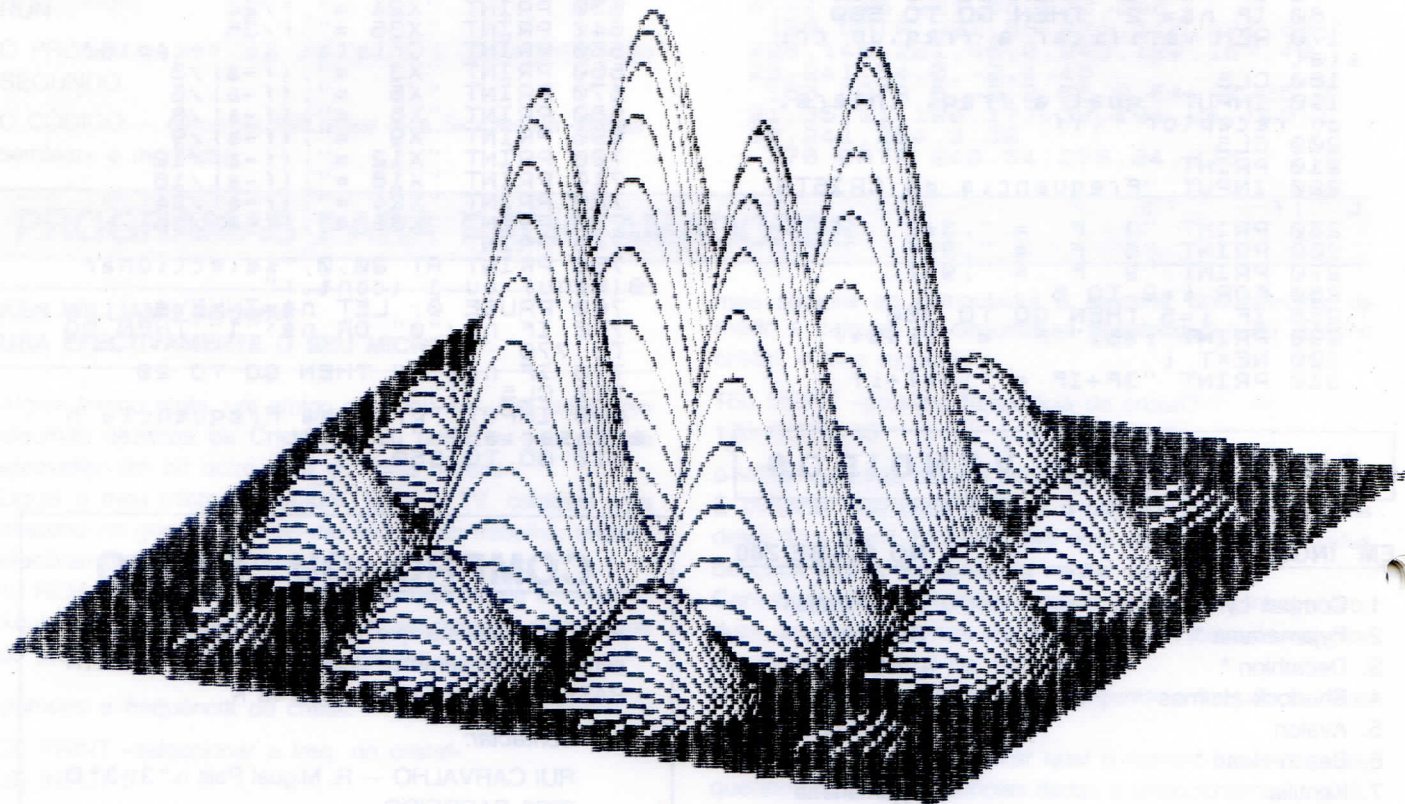
FUNÇÕES TRI-DIMENSIONAIS

NEWBRAIN

Autor: PAULO CASTELO
PORTO

Este programa é idêntico em objectivo ao programa apresentado no boletim n.º 22, de Julho, feito por Manuel Quinaz para o SPECTRUM. Este é diferente por trabalhar no NEWBRAIN e por as estruturas tridimensionais serem opacas.
A função a desenhar é definida na linha 150 e os seus parâmetros na linha 160; é aconselhável que o número de intervalos para o eixo dos YY (NY) seja menor ou igual a metade do número de intervalos para o eixo dos XX (NX); o tempo que leva a desenhar uma função é proporcional ao número total de intervalos e (menos significativamente) à complexidade da função (FNN).

```
150  DEF   FNN = (1-COS(X*PI))*(1-COS(Y*PI))*(1-COS(4*X*PI))*(1-COS(4*Y*PI))
160  LET  IX= 0 : IY= 0 : IZ= 0 : SX= 2 : SY= 2 : SZ=16 : NX=192 : NY=96
```

demora 2horas e 35 minutos

```
10 REM:  ==================================
20 REM:  |  FNN     V3.1         "*S:FNN"  |
30 REM:  |  Funcoes   Tri-Dimensionais     |
40 REM:  |=================================|
50 REM:  |  Paulo Castelo    OU.17/10/84   |
60 REM:  |  NEWBRAIN     SEIKOSHA GP-250X  |
70 REM:  ==================================
80 REM:
90
100 REM: === Iniciar ===
110 REM:
120  CLOSE#8 : CLOSE#11 : OPEN#0,4,"L141"
```

```
130  PRINT "INICIO" : OPEN#11,11,"219" : OPEN#8,9,"2400"
140  REM:
150  DEF   FNN = (1-COS(X*PI)) * (1-COS(Y*PI)) : REM: demora 12.50 min.s
160  LET   IX= 0 : IY= 0 : IZ= 0 : SX= 4 : SY= 4 : SZ= 4 : NX=64 : NY=32
170  REM:  ~~~limite inferior~~~  ~~~limite superior~~~  ~~~n.interv.~~~
180  REM:
190  T=(SZ-IZ)*1.73 : PLOT RANGE ( (SX-IX)*3/2 *641/640, T *220/219 )
200  REM:
210  XY=(SX-IX)/(SY-IY)/2 : ZY=(SZ-IZ)/(SY-IY)/2
220  D=SX-IX : T=T/7.7 : DIM S(NX*3/2), I(NX*3/2)
230  FOR M=0 TO NX*3/2+.1
240     S(M)=-1E99 : I(M)=1E99
250  NEXT M
260  PUT 31
290
300 REM: === Desenhar o grafico ===
310 REM:
320  FOR Y=IY TO SY +.1*(SY-IY)/NY STEP (SY-IY)/NY
330     CX=(Y-IY)*XY-IX : CZ=(Y-IY)*ZY-IZ : C0=0
340     FOR N=0 TO NX +.1
350        X=D*N/NX+IX : Z=FNN+CZ+T*(NX-N)/NX
360        C=0 : I=N+(CX+IX)*NX/D
370        IF Z>S(I) THEN C=1 : S(I)=Z
380        IF Z<I(I) THEN C=1 : I(I)=Z
390        M=C : C=C AND C0 : C0=M
400        PLOT COLOUR (C), MOVE (X+CX,Z)
410     NEXT N
420  NEXT Y
430  LINPUT ("QUER COPIAR (S) ? ") A$
440  IF A$="S" OR A$="s" GOSUB 500
450
460 END
490
500 REM: === Copiar o grafico ===
510 REM:
520 REM: Trabalha com uma impressora SEIKOSHA GP-250X
530 REM:   a "2400" bauds atraves do interface 9 para
540 REM:   evitar a formatacao do interface 8 que po-
550 REM:   dem modificar os graficos copiados.
560 REM:   So'copia graficos de 640x219:
570 REM:                     (modifique e aprenda).
580  PUT#8, 27,"L",2
590  FOR M=0 TO 79
600     PUT#8, 10, 27,"G",1,182
610     FOR N=PEN(7)+M+80*218 TO PEN(7)+M STEP-80
620        C=PEEK(N) : PUT#8, C,C
630     NEXT N
640  NEXT M
650  PUT#8, 10, 27,"L",3
660  RETURN
```

ROTINICES

SPECTRUM

Adapt.: J. MAGALHÃES/Clube Z80

Como forma de ultrapassar o problema de o Spectrum não dispor directamente de certas funções, temo-nos preocupado em publicar pequenas rotinas de grande utilização. Neste caso trata-se da função LISTO (BBC).

Esta rotina que apresentamos torna possível, também no Spectrum, uma listagem de qualquer programa em BASIC com todos os ciclos FOR-NEXT destacados das outras linhas. Pode ser colocada em qualquer zona da RAM. Os espaços que separam o número de linha e a linha do código podem ser alterados pelo comando POKE:

POKE 23728,2 — 2 espaços

POKE 23728,32 — 1 linha

```
  10 CLEAR 31999
  20 FOR a=32000 TO 32076
  30 READ d: POKE a,d
  40 NEXT a
  50 DATA 62,2,205,1,22,42,83,92
,237,91,75,92,235,167,237,82,200
,216,235,205,40,26,62,32,215,17,
176,92,26,254,0,40,8,245,62,32,2
15,241,61,24,244,35,35,35,126,25
4,235,32,4,235,52,52,235,126,254
,243,32,4,235,53,53,235,205,182,
24,205,55,25,126,254,13,32,226,3
5,24,188,187
  60 POKE 23728,2
  70 RANDOMIZE USR 32000
```

DESAFIO

HUGO ASSUMPÇÃO, DE LISBOA, propõe aos sócios do CLUBE Z80 o seguinte «desafio»:

«Para criar no sócio um espírito de descoberta das suas potencialidades inventivas ao nível da programação e levá-lo a usar mais a cabeça e o lápis do que os dedos e os reflexos, proponho este desafio que, concerteza, todos vão conseguir apesar da dificuldade.

Exemplo:

Criar um programa que permita calcular a área de um paralelogramo rectângulo inscrito numa circunferência, dados unicamente o raio desta e um dos lados do paralelograma.

Os sócios mandariam as respostas (programas) e o CLUBE publicaria as duas melhores. Mas os sócios na resposta teriam que colocar outra questão-desafio. QUE TAL ACHAM A IDEIA?

(Depois enviarei um programa suposto didáctico sobre o problema que propus)».

Hugo Assumpção

Muitos de vocês, especialmente os que não possuem Microdrive, se pretenderem gravar um programa, dados ou código máquina automaticamente (ou seja, saltando o «start tape then press any Key»), podem fazer:

— POKE 23736,181 antes do comando SAVE → SEM INTERFACE 1

— POKE 23744,181 antes do comando SAVE → COM INTERFACE 1

QUEM RESPONDE?

«Gostava que me explicassem o motivo porque no meu computador, geralmente a partir destas posições de memória até ao fim, ou seja até 65 535, quando eu ligo o computador estão ocupadas com estes números e nestas posições eu não posso alterar, porque o computador não aceita.»

Eduardo Pardinha — Leça da Palmeira

```
65300    62
65301    24
65302    71
65303    13
65304    243
65305    133
65306    206
65307    11
65308    228
65309    80
65310    206
65311    11
65312    229
65313    80
65314    28
65315    23
65316    220
65317    10
65318    206
65319    11
65320    235
65321    80
65322    22
65323    23
65324    220
65325    10
65326    215
65327    24
65328    177
65329    51
65330    25
65331    93
65332    5
65333    5
65334    219
65335    2
65336    219
65337    2
65338    77
65339    0
65340    1
65341    7
65342    0
65343    0
65344    0
65345    89
65346    84
65347    0
65348    101
65349    14
65350    192
```

QUANDO UM COMPUTADOR SE TORNA BILINGUE

— o compilador PASCAL H4T/Spectrum

Apesar dos seus 16 kbytes de BASIC residentes na memória ROM, o Spectrum não é rígido e pode adaptar-se a uma ou mais linguagens novas: por exemplo o PASCAL standard. Se o meu amigo possui um Spectrum com 48 kbytes de memória central, então 800$00 serão suficientes para aproveitar toda a potência do PASCAL.
Efectivamente a versão de PASCAL da HighSoft — 4T — introduz no seu Spectrum a linguagem PASCAL. O compilador e os programas de execução ocupam cerca de 20 kbytes e os restantes 11 kbytes podem ser usados para compilar grandes programas.
A cassette com o programa «Pascal — 4T —» comporta dois registos. O primeiro é um pequeno programa em BASIC que serve de interface entre o BASIC e o PASCAL.
O segundo, é um bloco de bytes correspondente ao código do compilador, do editor e dos programas de execução.
A documentação de 62 páginas compreende as indicações sobre a colocação em funcionamento do PASCAL 4T: sobre a sintaxe, explicações sobre os procedimentos e funções pré-definidas, diversas opções do compilador e do editor, bem como anexos, que incluem por exemplo pequenos programas.
Lamentamos que esta documentação seja escrita para adaptação deste compilador a vários tipos de máquinas, o que dificulta um pouco o entendimento do utilizador Spectroviano. Neste caso teremos, as particularidades do Spectrum um pouco desviadas da «massa» de informação contida no manual.
Após cerca de 2 minutos e 20 segundos de carregamento, podemos entrar directamente no Editor.
É necessário além disso, o endereço que designa a zona de memória em que queremos instalar o PASCAL. Poderá assim, reservar lugar na zona mais alta da memória, para aí escrever extensões pessoais aos programas de execução. Feita a inicialização, poderá começar a escrever programas nesta nova linguagem.
O Editor é fácil de usar. Compreende os comandos de inserção de texto, listagem de programas fonte, destruição ou duplicação de linhas e renumeração das mesmas.
A função FIND permite pesquisar uma string (cadeira de caracteres) no interior de um ficheiro fonte e de o substituir por outra string ou cadeia de caracteres por cada ocorrência da string original.
Outros comandos (numerosos) de edição de linhas facilitam o trabalho do programador.

APÓS A INICIALIZAÇÃO TERÁ UMA COMPILAÇÃO RÁPIDA

Os ficheiros fonte podem ser salvaguardados em fita magnética e podemos portanto carregar os programas quando necessitarmos deles.
Também é possível executar a fusão de vários programas fonte.
Um procedimento (sub-programa em PASCAL) pode ser guardado isoladamente e ser inserido isoladamente entre duas linhas de um programa fonte se ele contiver o pseudo-comentário de compilação INCLUDE.
A compilação é rápida. Existirá uma paragem desde que seja detectado um erro, no entanto, o utilizador está autorizado a retirar rapidamente a linha em causa no decurso da compilação ou editar e modificar.
Em anexo à documentação, existe uma lista em que são apresentados os 72 códigos de erro possíveis e respectiva correspondência (erros detectados durante a compilação). Outra lista apresenta os 11 códigos de erro possíveis durante a execução, tais como por exemplo, a ultrapassagem da capacidade ou uma divisão por zero.
Os parâmetros da compilação estão colocados em pseudo-comentários precedidos do caracter $.
As opções do compilador são seguidas de + ou de — segundo a opção esteja ou não activa.
Opção L — autoriza ou não a listagem do texto fonte e do endereço do código objecto.
Opção O — verifica ou não a ultrapassagem da capacidade das operações aritméticas (overflow).
Opção S — determina se o controlo da pilha é executado.
Opção A — assegura os limites definidos na secção de declaração das variáveis.
Opção P — permite obter a listagem da compilação na impressora.
Opção C — teremos um ganho de 6% sobre o funcionamento normal, dado que deixa de controlar periodicamente se a tecla de BREAK foi premida.
Podemos guardar em fita magnética o programa objecto (o código compilado), bem como os programas de execução, sem compilador, dispondo assim de um programa que se carregará simplesmente através do comando BASIC «load», e será executado automaticamente!
O programa de ordenação que em BASIC demora cerca de 3 minutos, ao ser traduzido para PASCAL será executado em 2 segundos (ordenar cerca de 100 números).
É preciso ver que os números tratados em BASIC são em vírgula flutuante, codificados em cinco bytes, enquanto que em PASCAL HS4T nós pedimos a ordenação dos números inteiros codificados em oito bytes.
Para tentar comparar coisas comparáveis, é preciso modificar o programa PASCAL para que ele possa executar a ordenação de uma centena de números reais codificados em quatro bytes.
Mesmo assim o tempo de ordenação será de 4 segundos.
Outro exemplo, este agora o de um programa invertendo os 45 000 pixel's do écran, ou seja imprimindo um ponto onde anteriormente ele não existia ou eliminahdo o ponto se ele existe no écran, confirmará a potência do PASCAL em relação ao BASIC.
O programa BASIC usa as funções POINT (x,y) e PLOT-OVER, realizando a operação em cerca de 10 minutos.
Em PASCAL usando os procedimentos POINT e PLOT (tal como estão descritas no modo de emprego do H4T) teremos o mesmo resultado em 1 minuto e 1/4, ou seja uma relação de tempo de 7.5.
Note-se que o mesmo programa escrito exclusivamente em linguagem máquina, terá uma duração de 8 segundos.

Em relação ao PASCAL standard encontraremos procedimentos e funções suplementares.
PEEK e POKE podem gerar vários bytes, contrariamente aos seus equivalentes BASIC.
TIN e TOUT permitem carregar ou gravar as variáveis na fita da cassette.
INP e OUT permitem comunicar com as portas de saída.

Os comandos tais como POINT, BORDER, SCREEN, etc. não estão disponíveis, mas podem ser introduzidos num procedimento graças à função INLINE.

Em conclusão, se descobriu o BASIC quando encontrou o material da Sinclair, aposte agora em descobrir uma linguagem potente e estruturada com o PASCAL HP4T.

Programa de ordenacao em BASIC - tratamento em 3 minutos

```
1 REM ******* ORDENAR **********
10 INPUT S
20 DIM T(S)
30 LET NN=S
40 LET N1=NN+1
50 FOR N=1 TO NN:LET T(N)=N1-N:NEXT N
60 PRINT "SERIE INICIAL"
70 GOSUB 500
80 FOR I=2 TO NN
90 FOR J=NN TO I STEP -1
100 IF T(J-1) > T(J) THEN LET X=T(J-1): LET
    T(J-1) = T(J): LET T(J)=X
110 NEXT J
120 NEXT I
130 PRINT TAB 0;"ORDENACAO -METODO TROCAR -"
140 GOSUB 500
150 STOP
500 FOR N=1 TO NN: PRINT T(N);" ";:NEXT N
510 RETURN
```

PROGRAMA DE ORDENACAO EM PASCAL - HP4T - tratamento em 2 segundos

```
AC53    10 PROGRAM ORDENAR
AC53    20 CONST NN=100
AC53    30 VAR TAB:ARRAY (1...100)
           OF INTEGER ;
AC5C    40 N,N1:INTEGER;
AC5C    50
AC5C    60
AC5C    70 PROCEDURE IMPTAB;
AC5F    80 VAR N:INTEGER;
AC5F    90 BEGIN
AC77   100 FOR N:=1 TO NN  DO
           WRITE (TAB (N),  '  ')
ACD1   110 END;
ACDB   120
ACDB   130
ACDB   140 PROCEDURE TROCA;
ACDE   150 VAR I,J,X:INTEGER;
ACDE   160
ACDE   170 BEGIN
ACF6   180 FOR I:=2 TO NN DO
AD19   190
AD19   200 BEGIN
AD1C   210 FOR J:=NN DOWNTO I DO
AD45   220 IF TAB (J-1)  > TAB (J)
AD89   230
AD89   240 THEN BEGIN
AD98   250 X:=TAB (J-1);
ADC0   260 TAB (J-1)  :=TAB(J);
AE05   270 TAB(J)  :=X
AE23   280 END
AE2E   290 END
AE32   300 END;
AE40   310
AE40   320
AE40   330 BEGIN
AE49   340 N1:=NN+1;
AE50   350 FOR N:=1  TO NN DO
AE6A   370 WRITELN ('SERIE INICIAL');
AEBB   380 IMPTAB;
AEC0   390 WRITELN;
AEC3   400 TROCA;
AEC8   410 WRITELN ('ORDENAR-METODO
           TROCAR-');
AEE5   420 IMPTAB
AEE5   430 END.
End Address : AEEC
Run ?
```

NOVOS PROGRAMAS

ENDURO

Jogo de perícia que consiste em ultrapassar o n.° de carros indicados no jogo, e para cada dia que passa, o n.° de carros a ultrapassar aumenta.
No primeira dia temos que ultrapassar 200 carros, no 2.° dia 300 carros, no 3.° dia 400 carros, e assim sucessivamente. Cada dia é constituído por 3 fases (Manhã, tarde), (anoitecer, noite), (amanhecer, manhã) em que cada uma delas vai apresentar maiores ou menores dificuldades. Pistas de gelo, nevoeiro, em que só se vêm os farolins traseiros dos carros, etc. ...... O n.° de carros é ilimitado, e o jogo acabará só quando se passar de um dia para outro sem que se tenha ultrapassado o n.° previsto de carros. O jogo indica-nos o n.° de carros que nos falta ultrapassar.

FRANK'N STEIN

Você é o cientista louco que quer montar o monstro, acoplando-lhe membros de outros seres humanos. Para isso, terá que andar a recolher os membros por ordem (crânio, costela direita, costela esquerda, etc....) e cuidado com os seres e objectos que o perseguem, assim como violentas descargas eléctricas, se pisar alguma lâmpada. Em cada nível existem molas que lhe permitirão, se se puser em cima delas, saltar para um nível superior. O grande problema é que FRANK'NSTEIN revolta-se contra o seu dono e criador, e através da passagem de vários cenários. Você terá sempre, que tentar desactivá-lo, o que nem sempre será fácil. O programa tem gráficos de alta definição e algumas dezenas de cenários diferentes.

PYJAMARAMA

Bem vindo ao pesadelo do mundo de Wall Week. Todas as noites, Wally tem um pesadelo para defrontar. A partir do momento em que substituis Wally não poderás dormir como uma pessoa normal.
Podes acordar como eu, e usar um relógio despertador; deste modo quando soar o despertador, desaparece o pesadelo.
Durante o pesadelo, terás que levar a cabo certas tarefas, que exigem a recolha de objectos, e em certos casos, apenas poderás recolher um se estiveres previamente na posse de um outro.
Apenas podes ser portador de dois objectos simultaneamente. Se aparece outro que não podes transportar, podes saltar por cima dele.
Existem diferentes salas na casa onde estás e a energia que possuis é um factor importante, pelo que para refazeres a tua energia terás de ir comer a diferentes salas. Possuis «3 vidas» e tens possibilidade de refazeres a tua energia 3 vezes. Poderás usar Joystick ou Teclado.

GIANT'S REVENGE

Após o programa estar instalado na máquina, terá de dar entrada a 4 cores codificadas. Se existir um erro até à terceira entrada, pode usar «S» e recomeçar.
Se três das entradas estão correctas e a num. 4 estiver errada o programa «vai ao ar» e terá de introduzir o programa de novo.
Trata-se de uma aventura, em que JACK vai pesquisar uma caverna, a partir de um buraco que JACK encontra no seu jardim.
Existem nessa caverna, imensos tesouros e o «Elixir da Vida» que JACK vai tirar ao Gigante. O programa é compatível com Joystick.

FACTORY BREAKOUT

Você é o ZIRKY, o último Robot que resta de uma fábrica e que está louco. O nosso objectivo é escapar através dos 3 cenários do jogo.
No primeiro objectivo é escapar através dos 3 cenários do jogo.
No primeiro cenário ZIRKY está numa cápsula-ovo e deve destruir os micro-raios que o rodeiam antes que lhe toquem. Quando a coluna de energia que está por baixo chegar até ele, é transportado para o tapete rolante que o conduzirá ao 2.° cenário.
No 2.° cenário, ZIRKY deve passar para o outro lado do cenário, sem ser atingido pelos raios dos conversores de energia.
Na 3.ª fase do jogo, encontrará uma chave para sair da fábrica, mas primeiro terá que passar pelas 12 portas que estão no seis níveis, tornando-as encarnadas e depois azuis (o que as fecha). Há 4 campos de forças que o ajudam a combater os monstros e a fechar as portas.

DAMAS

O tradicional jogo de damas agora para o Spectrum, 48K ou 16K, funciona com os movimentos normais de acordo com as normas que os portugueses usam ou seja a DAMA anda quantas casas quiser.

PREÇO DE CADA JOGO: 400$00

MERCADO Z80

O MERCADO Z80 É UMA SECÇÃO DO CLUBE Z80 QUE EMPRESA AOS SEUS SÓCIOS PROGRAMAS E LIVROS (SPECTRUM) PARA CONSULTA E MELHOR CONHECIMENTO/APROVEITAMENO DE MICROCOMPUTADORES.

— PROGRAMAS: Todos os jogos e utilitários que existem no CLUBE Z80, excepto programas de cópia e programas com direitos de autor (Ex.: "Cálculo de Estruturas").

— LIVROS: Cerca de 40 títulos diferentes.

COMO TORNAR-SE SÓCIO DO MERCADO Z80?

Para poder ter em sua posse 5 cassetes ou livros durante um mês, basta enviar-nos um depósito de Esc. 2000$00 (garantia de que os materiais nos serão devolvidos em estado de conservação e funcionamento idêntico àquele em que foram enviados).
Ao mesmo tempo, deverá remeter-nos a quantia de Esc. 1000$00 que será a base da sua "Conta-Corrente". Essa quantia servirá para pagar as suas despesas:

— Taxa de utilização dos produtos: 250$00 (referente a 5 unidades, entre livros e programas).

— Instruções dos programas (no caso de o sócio não as devolver, debitar-lhe-emos 5$00 por folha).

— Embalagem Postal: 20$00 a 30$00 (no caso de o pedido ser feito via CTT.

— Portes dos CTT's: 40$00 a 80$00 (no caso de o pedido ser feito via CTT.

IMPORTANTE!

— O depósito de 2000$00 pertence integralmente ao sócio desde que os materiais por ele utilizados nos sejam devolvidos nas mesmas condições em que saíram do CLUBE Z80. Assim, quando o sócio desistir do MERCADO Z80, essa quantia ser-lhe-á entregue.
Em caso de extravio, danos ou avarias dos materiais, o sócio pagará o valor comercial dos respectivos produtos (a descontar no depósito de 2000$00).

— No caso de os produtos seguirem via CTT, o sócio não pagará para levantar a encomenda. As despesas serão pagas por nós, no momento da expedição, e debitadas ao sócio (a descontar no depósito de 1000$00).

— Quando as suas despesas estiverem a atingir os 1000$00 avisá-lo-emos, e o sócio deverá renovar essa quantia de modo a cobrir despesas seguintes.

— A taxa de utilização dos produtos é fixa — 250$00. Ela refere-se ao conjunto de 5 unidades. (Pagará sempre 250$00 mesmo que peça só uma unidade).

QUE PRODUTO E QUE QUANTIDADES?

O sócio nunca pode pedir mais do que 5 unidades de cada vez (entre livros e cassetes). Quanto a livros não poderemos empresar mais do que um. Assim, o sócio poderá pedir:

— 5 cassettes
ou
— 4 cassettes + 1 livro

QUAL O TEMPO DE UTILIZAÇÃO?

O sócio poderá ficar com os produtos durante 1 MÊS, no máximo. Findo esse período, deverá devolvê-los ao CLUBE Z80.
O MERCADO Z80 só atenderá dois pedidos por mês, para cada sócio.

COMO FAZER O PEDIDO?

Numa carta, escreva pelo menos 10 títulos (por ordem de prioridade). Se os 5 primeiros não estiverem disponíveis, enviaremos os outros evitando grandes esperas de produtos que estejam em circulação.
Ao devolver os produtos, inclua uma carta com o pedido seguinte.

IMPORTANTE! O SÓCIO SÓ PODERÁ EFECTUAR UM NOVO PEDIDO JUNTAMENTE COM A DEVOLUÇÃO DO MATERIAL CORRESPONDENTE AO PEDIDO ANTERIOR (ou depois, se preferir).

Se estiver interessado no MERCADO Z80, faça já o seu 1.º pedido, enviando 3000$00 e o cupão abaixo devidamente preenchido.

INSCRIÇÃO NO MERCADO Z80

NOME ______________________

ENDEREÇO ______________________

CÓDIGO POSTAL ______________________

TELEFONE ____________

ENVIO 3 000$00 (2 000$00 como garantia de que devolverei os produtos em boas condições + 1 000$00 para a minha «Conta Corrente» em:

☐ Cheque n.º ____________ ☐ Vale Postal ☐ Dinheiro

Banco ____________ N.º ____________ ____________

Data ___/___/___ *Assinatura do Sócio* ______________________

É SÓCIO DO CLUBE Z80? ☐ SIM ☐ NÃO

CLUBE Z80

INSCRIÇÃO COMO ASSOCIADO

O **CLUBE Z80** está aberto a todos os utilizadores de microcomputadores.

A intenção de associar os entusiastas das micro-máquinas, é exclusivamente a de permitir:

1 — PUBLICAÇÃO DE UM JORNAL MENSAL, onde sejam publicados programas de uso geral ou específico como no caso da educação.

2 — PROMOVER TROCAS DE PROGRAMAS, e trocas de experiências; tanto no caso do Software (programação), como no caso do Hardware (electrónica).

3 — PROMOVER DESCONTOS NA AQUISIÇÃO DE PROGRAMAS.

4 — LANÇAR CURSOS DE PROGRAMAÇÃO EM BASIC — PASCAL OU OUTRAS LINGUAGENS E DIVULGAR O USO DE LINGUAGEM MÁQUINA.

NOME ....

IDADE .... COMPUTADOR TIPO ....

PROFISSÃO ....

ENDEREÇO ....

.... TELEF. ....

ASSINATURA ANUAL — Esc. 1 500$00 ☐

ASSINATURA SEMESTRAL — Esc. 750$00 ☐

CHEQUE OU VALE DO CORREIO

N.º ....

BANCO ....

DATA ..../......../........

JÁ SÓCIO ☐

NOVO SÓCIO ☐ → A partir do mês de .... (inclusive)